

On the decidability of the Σ_2 theories of the arithmetic and hyperarithmetic degrees as uppersemilattices

James Barnes

Abstract

We establish the decidability of the Σ_2 theory of both the arithmetic and hyperarithmetic degrees in the language of uppersemilattices i.e. the language with $\leq, 0$ and \sqcup . This is achieved by using Kumabe-Slaman forcing - along with other known results - to show that given finite uppersemilattices \mathcal{M} and \mathcal{N} , where \mathcal{M} is a subuppersemilattice of \mathcal{N} , then for both degree structures, every embedding of \mathcal{M} into the structure extends to one of \mathcal{N} iff \mathcal{N} is an end-extension of \mathcal{M} .

Introduction

Given a transitive notion r of reducibility between subsets of ω (we write $X \leq_r Y$ to mean X is r -reducible to Y), the r degrees (denoted \mathcal{D}_r) is the quotient of 2^ω under the equivalence relation \equiv_r of mutual reducibility: $X \equiv_r Y$ iff $X \leq_r Y$ and $Y \leq_r X$. The relation \leq_r induces a partial order on \mathcal{D}_r , and given $X \subseteq \omega$, we call its equivalence class in \mathcal{D}_r its r degree. The primary such notion is Turing reducibility: $X \leq_T Y$ if there is a Turing machine with oracle Y that computes the characteristic function of X .

Two related notions are arithmetic and hyperarithmetic reducibility (denoted \leq_a and \leq_h respectively). We say X is arithmetic in Y if X is Turing reducible to the n th jump of Y for some $n \in \omega$ (the jump Y' of Y is the set $\{e : \{e\}^Y(e) \downarrow\}$, the n th jump is the n th iterate of the jump operator). Equivalently, X is arithmetic in Y if X is definable in first-order arithmetic with a predicate for membership in Y . X is hyperarithmetic in Y if X is Turing reducible to the α th jump of Y for some Y -recursive ordinal α (at limit levels one takes effective unions). Equivalently, X is hyperarithmetic in Y if X is Δ_1^1 in Y in second-order arithmetic.

The three degree structures $\mathcal{D}_T, \mathcal{D}_a, \mathcal{D}_h$ all have a least element: the degree of the empty set. Furthermore, each has a natural join operator induced by the operation \oplus on 2^ω , which is defined by

$$X \oplus Y = \{2n : n \in X\} \cup \{2n + 1 : n \in Y\}.$$

It is easy to check that the degree of $X \oplus Y$ depends only on the degree of X and Y , and that it is their least upperbound. The reducibility, the 0 degree, and the join operator endow each of these classes of degrees with the structure of an uppersemilattice with 0. (We abbreviate this as USL).

1 Preliminaries

A major project in recursion theory has been to classify the dividing line between the decidable and undecidable fragments of the theories of degree structures in various languages. In this paper we extend the result of Jockusch and Slaman [3] that the Σ_2 theory of \mathcal{D}_T as a USL is decidable, to \mathcal{D}_a and \mathcal{D}_h . This is sharp, in the sense that the Σ_3 theory of all three structures is undecidable, even with only \leq in the language. This follows for any USL \mathcal{L} into which every finite USL embeds as an initial segment. The result is originally due to Schmerl for \mathcal{D}_T , see Lerman - Thm VII.4.5 [5] for a proof. For the initial segment results in \mathcal{D}_a and \mathcal{D}_h , see Simpson [10], and Kjos-Hanssen and Shore [4] respectively.

Our approach follows that of Jockusch and Slaman, who showed that given a USL \mathcal{L} , the Σ_2 theory of \mathcal{L} is decidable if the following three facts obtain:

- (1) Every finite USL embeds as an initial segment of \mathcal{L} .
- (2) For every $x \in \mathcal{L}$ and $n > 0$ there are $y_0, \dots, y_n \in \mathcal{L}$ which are mutually incomparable over x (i.e. for each $i = 0, \dots, n$ the points y_i and $x \sqcup \bigsqcup_{j \neq i} y_j$ are incomparable in \mathcal{L}).
- (3) For any $a, b, c_i, d_i, e_i \in \mathcal{L}$,

$$\left[\bigwedge_{i \in \omega} d_i \not\leq c_i \ \& \ (a \not\leq c_i \text{ or } d_i \not\leq c_i \sqcup b) \right] \rightarrow (\exists g) \left[b \leq a \sqcup g \ \& \ \bigwedge_{i \in \omega} d_i \not\leq c_i \sqcup g \ \& \ \bigwedge_{i \in \omega} g \not\leq e_i \right]$$

We call (3) the extended Posner-Robinson theorem for \mathcal{L} (for the decidability result, one actually only needs the above to hold for arbitrarily long finite conjunctions, which is slightly weaker than allowing countable conjunctions).

The analogues of (1) and (2) both hold in \mathcal{D}_a and \mathcal{D}_h . Citations for (1) were given above, (2) follows by taking the columns of a sufficiently generic Cohen real (relativized to a suitable degree). (See Odifreddi [6] for an exposition of arithmetical Cohen forcing, and Sacks IV.3 [7] for hyperarithmetical Cohen forcing). Thus, the question of the decidability of the Σ_2 theory of \mathcal{D}_a and \mathcal{D}_h can be resolved by establishing the extended Posner-Robinson theorem for these structures. The rest of this paper is devoted to showing that this is indeed the case.

Our method is a forcing construction, however, is different from that of Jockusch and Slaman. They devise a forcing, depending on representatives $A \in \mathbf{a}$ and $B \in \mathbf{b}$, coding $B(n)$ into the top element of the n th column of their generic g . This renders $B \leq_T g'$, and so B is both arithmetic and hyperarithmetical in g . If there are (arithmetic or hyperarithmetical) degrees $\mathbf{c}_i, \mathbf{d}_i$ such that $\mathbf{d}_i \leq \mathbf{c}_i \sqcup \mathbf{b}$ and $\mathbf{a} \not\leq \mathbf{c}_i$, then $d_i \leq c_i \sqcup g$, and so such a forcing does not suffice in our settings.

Instead, we use Kumabe-Slaman forcing, in particular, the presentation given in Shore and Slaman [9]. In the hyperarithmetical setting we also have the added concern of preserving $\omega_1^{\mathbf{c}_i}$ for each \mathbf{c}_i .

2 The notion and language of forcing

Definition 2.1 (Turing functionals). For $x \in \omega, y \in \{0, 1\}$, and $\sigma \in 2^{<\omega}$, we call the tuple $\langle x, y, \sigma \rangle$ an **axiom** or a **computation**. A **Turing functional** is a set Φ of computations such that, if σ_1 and σ_2 are compatible, and $\langle x, y_1, \sigma_1 \rangle, \langle x, y_2, \sigma_2 \rangle \in \Phi$, then $y_1 = y_2$ and $\sigma_1 = \sigma_2$.

Furthermore, Φ is **use-monotone** if,

- whenever $\sigma_1 \subsetneq \sigma_2$ and $\langle x_1, y_1, \sigma_1 \rangle, \langle x_2, y_2, \sigma_2 \rangle \in \Phi$, then $x_1 < x_2$, and
- whenever $x_1 < x_2$ and $\langle x_2, y_2, \sigma_2 \rangle \in \Phi$, there is a y_1 and a $\sigma_1 \subsetneq \sigma_2$ such that $\langle x_1, y_1, \sigma_1 \rangle \in \Phi$.

Notation. We write $\Phi(x, \sigma) = y$ to mean there is a $\tau \subseteq \sigma$ such that $\langle x, y, \tau \rangle \in \Phi$. If $X \subseteq \omega$, we write $\Phi(x, X) = y$ to mean there is an n such that $\Phi(x, X \upharpoonright n) = y$, and write $\Phi(X)$ for the (partial) function evaluated in this way. Note that $\Phi(X)$ is (uniformly) recursive in the join of X and Φ .

Definition 2.2 (Kumabe-Slaman forcing). **Kumabe-Slaman forcing** is the partial order \mathcal{P} , with conditions $p = (\Phi_p, \mathbf{X}_p)$, where Φ_p is a finite use-monotone Turing functional and \mathbf{X}_p is a finite set of subsets of ω (hereafter we refer to such $X \subseteq \omega$ as **reals**). For elements p, q of \mathcal{P} , we say $q \leq_{\mathcal{P}} p$ (q **extends** p) if

- $\Phi_p \subseteq \Phi_q$ and for all $(x_q, y_q, \sigma_q) \in \Phi_q \setminus \Phi_p$ and all $(x_p, y_p, \sigma_p) \in \Phi_p$ the length of σ_q is greater than the length of σ_p .
- $\mathbf{X}_p \subseteq \mathbf{X}_q$.
- For every x, y , and $X \in \mathbf{X}_p$, if $\Phi_q(X)(x) = y$, then $\Phi_p(X)(x) = y$ i.e. q can't add new axioms $\langle x, y, \sigma \rangle$ where $\sigma \subset X$ for any $X \in \mathbf{X}_p$.

We think of a forcing condition $p = (\Phi_p, \mathbf{X}_p)$ as approximating a (possibly nonrecursive) Turing functional Φ_G . To a descending sequence of conditions $(p_n)_{n=1}^\infty$ we associate the object $\Phi_G = \bigcup_n \Phi_{p_n}$, a use-monotone Turing functional.

On a formal level, we code each finite use-monotone Turing functional Φ as a natural number, which encodes each element of Φ and also the size of Φ . For instance, given some recursive encoding of triples $\langle x, y, \sigma \rangle$ as positive integers, we could code $\Phi = \{\langle x_1, y_1, \sigma_1 \rangle, \dots, \langle x_n, y_n, \sigma_n \rangle\}$ as the product $\prod_{i=1}^n P(i)^{\langle x_i, y_i, \sigma_i \rangle}$ where $P(i)$ is the i th prime. Consequently, a forcing condition is coded as a pair, the first coordinate is a natural number and the second a finite set of reals. We dub the first coordinate the **finite part**, and the second the **infinite part**. Given finite use-monotone Turing functionals Φ_p and Φ_q , we will occasionally abuse notation and write $\Phi_p \leq_{\mathcal{P}} \Phi_q$, by which we mean $(\Phi_p, \emptyset) \leq_{\mathcal{P}} (\Phi_q, \emptyset)$.

We will often have occasion to modify our notion of forcing somewhat, either by insisting that certain axioms don't belong to the finite part, or the reals can only come from a particular subclass of 2^ω . For the first modification we introduce some notation:

Notation. Given reals A and B (think of representatives from \mathbf{a} and \mathbf{b} respectively), we say that a Turing functional Φ **partially computes** B **on input** A if whenever $\Phi(A)(n)$ is defined, it equals $B(n)$. More explicitly, if $\langle x, y, \sigma \rangle \in \Phi$ is an axiom such that $\sigma \subseteq A$ (we say this axiom **applies to** A), then $y = B(x)$.

Definition 2.3 (Restricted Kumabe-Slaman forcing). $\mathcal{Q}_{A,B}$ is the subset of \mathcal{P} containing all those conditions $q = (\Phi_q, \mathbf{X}_q)$ such that Φ_q partially computes B on input A , and \mathbf{X}_q does not contain A . Extension in $\mathcal{Q}_{A,B}$ (denoted $\leq_{\mathcal{Q}_{A,B}}$) is defined as for \mathcal{P} . The partial order $\mathcal{Q}_{A,B}$ is a suborder of \mathcal{P} , and will be abbreviated as \mathcal{Q} if it is understood what A and B are. We employ a similar convention as for \mathcal{P} and write $\Phi_q \leq_{\mathcal{Q}} \Phi_r$ to mean $(\Phi_q, \emptyset) \leq_{\mathcal{Q}} (\Phi_r, \emptyset)$.

Our main goal is to construct a sufficiently generic sequence of conditions into which we code B via A . More precisely, for each $n \in \omega$, our generic will have an axiom $(n, B(n), \alpha)$ where α is an initial segment of A . This will ensure that $\Phi_G(A)$ is the characteristic function of B , and so $B \leq_T \Phi_G \oplus A$. We will need to show that we can construct a generic sequence without adding “incorrect” axioms applying to A , and without adding A into the infinite part of any condition (which would prevent us from adding any axioms about A later).

The forcing $\mathcal{Q}_{A,B}$ has all this hard-wired in, however, it is at the expense of checking whether Φ can serve as the finite part of the condition being $A \oplus B$ recursive, instead of plain old recursive. For \mathcal{P} , we need to show that we can manually avoid adding in bad axioms or A into the infinite part of a condition, while still constructing a generic sequence.

We ensure that $\mathbf{d}_i \not\leq \mathbf{c}_i \sqcup g$ via genericity. At a high level, our method is quite standard, we will define a forcing language that will be powerful enough to express all the possible pertinent reductions from $C_i \oplus \Phi_G$ to D_i (for chosen representatives $C_i \in \mathbf{c}_i$ and $D_i \in \mathbf{d}_i$). We will try to diagonalize against each of these reductions in turn. The result for the arithmetic degrees is a direct corollary of our methods for the hyperarithmetic degrees, so we need not treat them separately.

For forcing in the arithmetic or Turing degrees, a forcing language based upon first-order arithmetic often suffices. However, for the hyperarithmetic setting something more expressive is required. We use recursive infinitary formulas as presented in Ash and Knight chapter 9 [1] (over the language of arithmetic, see Shore [8] for an exposition). Roughly speaking, at the ground level we have the quantifier-free finitary sentences in our language, and then, for each recursive ordinal α , a Σ_α^r formula is an r.e. disjunction of the form

$$\bigvee_i (\exists \vec{u}_i) \psi_i(\vec{u}_i, \vec{x})$$

where each ψ_i is a $\Pi_{\beta_i}^r$ formula for some $\beta_i < \alpha$. Similarly a Π_α^r formula is an r.e. conjunction of universally quantified formulas all of which are $\Sigma_{\beta_i}^r$ for some $\beta_i < \alpha$.

Definition 2.4 (The language of forcing). We start with \mathcal{L} , the set containing the following nonlogical symbols:

- Numerals: $\underline{0}, \underline{1}, \underline{2}, \dots$

- A 5-ary relation for the universal recursive predicate: $\varphi(e, x, s, y, \sigma)$ (to be interpreted as, the e th Turing machine on input x running for s steps with oracle σ halts, and outputs y , we will often write this as $\{e\}_s^\sigma(x) \downarrow = y$, and write $\{e\}^X(x)$ for the partial function evaluated this way).
- A predicate for a string being an initial segment of our generic: $\sigma \subset \Phi_{\mathbf{g}}$

From \mathcal{L} we build up the recursive infinitary formulas as indicated in Ash and Knight, with the convention that the ground level formulas are in CNF, and denote this collection $\mathcal{L}_{\omega_1, \omega}^r$.

To relativise to some $S \subseteq \omega$, we add a predicate (written $\sigma \subset \mathbf{S}$) for a string being an initial segment of S , and build up the S -recursive infinitary formulas i.e. we allow S -r.e. infinitary conjunctions and disjunctions. We denote the collection of such formulas $\mathcal{L}_{\omega_1, \omega}^r(S)$ and rank the formulas as $\Sigma_\alpha^r(S)$ and $\Pi_\alpha^r(S)$ for $\alpha < \omega_1^S$.

We think of numerals as having a dual role in our language: they represent both numbers (in the obvious way) and also strings via some fixed recursive bijection between $2^{<\omega}$ and ω . We will suppress the technical apparatus needed, and will use lowercase Roman letters when we are thinking in terms of numbers, and lowercase Greek letters when we are thinking in terms of strings.

We can now define the forcing relation $\Vdash_{\mathcal{P}}$ for \mathcal{P} , with some $S \subseteq \omega$ as our set parameter.

Definition 2.5 (The forcing relation). The relation $\Vdash_{\mathcal{P}}$ between elements of \mathcal{P} and sentences of $\mathcal{L}_{\omega_1, \omega}^r(S)$ is defined by induction on the ordinal rank and complexity of the formula.

- (1) $p \Vdash_{\mathcal{P}} \underline{n} = \underline{m}$ iff $n = m$.
- (2) $p \Vdash_{\mathcal{P}} \varphi(\underline{e}, \underline{x}, \underline{s}, \underline{y}, \underline{\sigma})$ iff $\{e\}_s^\sigma(x) \downarrow = y$.
- (3) $p \Vdash_{\mathcal{P}} \underline{\sigma} \subset \mathbf{S}$ iff $\sigma \subset S$.
- (4) $p \Vdash_{\mathcal{P}} \underline{\sigma} \subset \Phi_{\mathbf{g}}$ iff for all n less than $|\sigma|$:
 - (a) If $\sigma(n) = 1$, then $n \in \Phi_p$.
 - (b) If $\sigma(n) = 0$, then either n is not of the form $\langle x, y, \tau \rangle$, or $n = \langle x, y, \tau \rangle$, $n \notin \Phi_p$ and
 - (i) there is an $\langle x_0, y_0, \tau_0 \rangle \in \Phi_p$ such that $|\sigma_0|$ is greater than $|\sigma|$, or, x_0 is greater than x and σ_0 is compatible with σ , or
 - (ii) σ is an initial segment of one of the elements of \mathbf{X}_p .
- (5) For an atomic sentence ψ , $p \Vdash_{\mathcal{P}} \neg\psi$ iff there is no $q \leq_{\mathcal{P}} p$ with $q \Vdash \psi$.
- (6) For sentences ψ_1, \dots, ψ_n that are literals, $p \Vdash_{\mathcal{P}} \psi_1 \vee \dots \vee \psi_n$ iff for all $q \leq_{\mathcal{P}} p$ there exists an $r \leq_{\mathcal{P}} q$ and an $i \in \{1, \dots, n\}$ such that $r \Vdash \psi_i$.
- (7) For sentences ψ_1, \dots, ψ_n that are finite disjunctions of literals, $p \Vdash_{\mathcal{P}} \psi_1 \wedge \dots \wedge \psi_n$ iff $p \Vdash \psi_i$ for each $i \in \{1, \dots, n\}$.

(8) $p \Vdash_{\mathcal{P}} \bigvee_i (\exists \vec{u}_i) \psi_i(\vec{u}_i)$ iff there is some i and some \vec{n} such that $p \Vdash_{\mathcal{P}} \psi_i(\vec{n})$.

(9) $p \Vdash_{\mathcal{P}} \bigwedge_i (\forall \vec{u}_i) \psi_i(\vec{u}_i)$ iff for each i, \vec{n} and $q \leq_{\mathcal{P}} p$ there is some $r \leq_{\mathcal{P}} q$ such that $r \Vdash \psi_i(\vec{n})$.

Note. We note some abuses of the language of forcing which we will employ hereafter. Firstly, we will stop underlining numerals. Secondly, although our language only has negations of atomic formulas, there is an S -recursive function neg that takes (an index for) a formula ψ and returns (an index for) a formula that is logically equivalent to the negation of ψ . Also, neg preserves the ordinal rank of the formula i.e. if ψ is $\Pi_{\alpha}^r(S)$, then $\text{neg}(\psi)$ is $\Sigma_{\alpha}^r(S)$, and visa versa. Consequently, we write $\neg\psi$ freely as if negation were a part of the language.

As expected, it can be shown that, for each S -recursive ordinal α , the claim that the partial function $\{e\}^{(S \oplus \Phi_G)^{(\alpha)}}$ is defined on an input x (and, optionally, has value y) can be expressed as a $\Sigma_{\alpha+1}^r$ sentence in our language, uniformly in e, x and y .

The definition of $\Vdash_{\mathcal{Q}}$ is analogous except in clause 4. In particular we must add a third possibility to (b), namely, that $\tau \subset A$ but $B(x) \neq y$.

Standard lemmas include that extension preserves forcing, no condition forces a sentence and its (formal) negation, and given a sentence and a condition, we can find an extension of that condition deciding the sentence.

The countability of $\mathcal{L}_{\omega_1, \omega}^r(S)$ implies that generic sequences exist. To a generic sequence $\{p_n\}$ we associate the generic object $\Phi_G = \bigcup \Phi_{p_n}$, which is the intended interpretation of $\Phi_{\mathcal{G}}$. With this, one proves the final standard lemma: truth equals forcing. Proofs of these standard lemmas can be found in chapter 10 of Ash and Knight.

Now a technical lemma about the complexity of the forcing relation.

Lemma 2.6. *Given a finite use-monotone Turing functional Φ_0 , and a quantifier-free finitary sentence ψ in $\mathcal{L}_{\omega_1, \omega}^r(S)$ (i.e. a Π_0^r sentence), we can S -recursively decide whether there is an \mathbf{X} such that $(\Phi_0, \mathbf{X}) \in \mathcal{P}$ forces ψ , uniformly in both the sentence and the functional.*

Furthermore, we can also make this decision for $\mathcal{Q}_{A,B}$ (and $\Vdash_{\mathcal{Q}}$), however, the procedure is uniformly recursive in $A \oplus B \oplus S$.

Proof. We begin with the claim for \mathcal{P} . By convention, finitary sentences are in conjunctive normal form. There are four types of atomic sentences

(1) $n = m$

(2) $\varphi(e, x, s, y, \sigma)$

(3) $\sigma \subset \mathbf{S}$

(4) $\sigma \subset \Phi_{\mathcal{G}}$

then there are the negations of these atoms, finite disjunctions of such literals, and finite conjunctions of such disjunctions.

For the first three types of atoms, a condition forces it iff it is true, and we can recursively in S check each of these facts. Similarly if one of these atoms is false, then no condition forces it to be true, and so every condition forces the negation.

The fourth type of atom is $\sigma \subset \Phi_G$. We claim that there is an \mathbf{X} so that $(\Phi_0, \mathbf{X}) \Vdash \sigma \subset \Phi_G$ iff for each $n < |\sigma|$, we have $\Phi_0(n) = \sigma(n)$ (which is uniformly recursive to check). To see this, note that a condition p forces $\sigma \subset \Phi_G$ iff $\sigma(n) = \Phi_p(n)$ and for each n such that $\sigma(n) = 0$, if $n = \langle x, y, \tau \rangle$, then $\langle x, y, \tau \rangle$ is prevented from entering any extension of p (i.e. there is either an axiom $(x_0, y_0, \tau_0) \in \Phi_p$ with $|\tau_0| > |\tau|$, or $x_0 \geq x$ and τ_0 and τ are compatible, or τ is an initial segment of some $X \in \mathbf{X}_p$).

Therefore, for p to force $\sigma \subset \Phi_G$, it is necessary that Φ_p agrees with σ everywhere that σ is defined. If Φ_0 and σ do agree, then for each $n < |\sigma|$ with $\sigma(n) = 0$ and n of the form $\langle x, y, \tau \rangle$, we can pick a real X extending τ and let \mathbf{X} be the collection of such reals. Then $(\Phi_0, \mathbf{X}) \Vdash \sigma \subset \Phi_G$ which completes the claim.

Now we turn to literals of the form $\neg(\sigma \subset \Phi_G)$. We claim that we can pick an \mathbf{X} to guarantee that $(\Phi_0, \mathbf{X}) \Vdash \neg(\sigma \subset \Phi_G)$ iff there is an $n < |\sigma|$ such that $\sigma(n) \neq \Phi_0(n)$. From the definition of forcing, a condition p forces $\neg(\sigma \subset \Phi_G)$ iff every extension of p fails to force $\sigma \subset \Phi_G$. Suppose Φ_0 and σ agree wherever σ is defined, then, by the above, we can pick \mathbf{X}_0 so that $(\Phi_0, \mathbf{X}_0) \Vdash \sigma \subset \Phi_G$. Therefore, for each \mathbf{X} we have that $(\Phi_0, \mathbf{X} \cup \mathbf{X}_0) \leq_P (\Phi_0, \mathbf{X})$ and $(\Phi_0, \mathbf{X} \cup \mathbf{X}_0) \Vdash \sigma \subset \Phi_G$. Consequently, if Φ_0 and σ agree wherever σ is defined, then, for each choice of \mathbf{X} there is an extension of (Φ_0, \mathbf{X}) forcing $\sigma \subset \Phi_G$. By taking the contrapositive we have established one direction of the claim.

For the other direction, suppose there is an n so that $\sigma(n) \neq \Phi_0(n)$. For such an n , either $\sigma(n) = 0$ and $\Phi_0(n) = 1$, or, $\sigma(n) = 1$ and $\Phi_0(n) = 0$. In the first case, no extension of (Φ_0, \emptyset) can force $\sigma \subset \Phi_G$ as n is in every stronger condition. Hence, $\mathbf{X} = \emptyset$ suffices as the set of reals. In the second case, if $n \neq \langle x, y, \tau \rangle$, then n is not an element of any Turing functional and so again $\mathbf{X} = \emptyset$ suffices. So suppose $n = \langle x, y, \tau \rangle$ and let X be any real extending τ . As $\langle x, y, \tau \rangle \notin \Phi_0$, it is also not in any extension of $(\Phi_0, \{X\})$, so no extension of this condition can force $\sigma \subset \Phi_G$, and so, $\{X\}$ is our set of reals, which completes the proof of the claim.

Now, suppose we have a sentence $\psi = l_1 \vee \dots \vee l_n$, which is a finite disjunction of literals. From the definition of forcing, it is clear that if a condition forces one of the disjuncts, it forces the whole disjunction. So, using induction, ask whether there is a literal l_i and a set of reals \mathbf{X} so that $(\Phi_0, \mathbf{X}) \Vdash l_i$. If there is, we are done, as such an \mathbf{X} suffices. If not, we claim no \mathbf{X} works. To see this, firstly, note that we get to ignore any of the literals with atomic parts of type (1), (2) or (3), as (\emptyset, \emptyset) decides these sentences as false. So now assume ψ is a disjunction of literals built from atoms of type (4). For each positive literal $\sigma_i \subset \Phi_G$, because we can't pick an \mathbf{X} to force it, then, Φ_0 and σ_i disagree somewhere on the domain of σ_i . Hence, for each i , there is a finite collection \mathbf{X}_i so that $(\Phi_0, \mathbf{X}_i) \Vdash \neg(\sigma_i \subset \Phi_G)$. Similarly, for each negative literal, as we can't pick an \mathbf{X} to force it, then, Φ_0 and σ_i agree on the domain of σ_i , and so, we can pick \mathbf{X}_i so that $(\Phi_0, \mathbf{X}_i) \Vdash \sigma_i \subset \Phi_G$.

So fix a set of reals \mathbf{X} , and define $\mathbf{X}_0 = \bigcup_{i=1}^n \mathbf{X}_i$. Then $(\Phi_0, \mathbf{X}_0 \cup \mathbf{X}) \leq_P (\Phi_0, \mathbf{X})$ yet $(\Phi_0, \mathbf{X}_0 \cup \mathbf{X})$ forces the opposite of each literal in ψ , i.e. forces $\neg\psi$. Consequently, (Φ_0, \mathbf{X}) does not force ψ for any \mathbf{X} .

For conjunctions $\psi_1 \wedge \dots \wedge \psi_m$, the definition of forcing is you force the conjunction iff you force each of the conjuncts. Inductively, ask if there are reals \mathbf{X}_i such that $(\Phi_0, \mathbf{X}_i) \Vdash \psi_i$, if “yes” in each case, then, $(\Phi_0, \bigcup_{i=1}^m \mathbf{X}_i)$ forces each conjunct. If any of them return a “no”, then no choice of reals can suffice for all conjuncts. This completes the decision procedure, uniformity is clear.

For \mathcal{Q} , only minor changes are needed. Firstly, one must check whether Φ_0 correctly computes B on input A , which is recursive in $A \oplus B$. Secondly, although the definition of $\Vdash_{\mathcal{Q}}$ for formulas of the form $\sigma \subset \Phi_{\mathbf{g}}$ is slightly different i.e. an axiom $\langle x, y, \sigma \rangle$ where $\sigma \subset A$, yet $y \neq B(x)$ is not a member of any \mathcal{Q} condition, this is, again $A \oplus B$ recursive to check. Finally, whenever we picked a real X to prevent an axiom from entering any stronger condition, X just had to extend some finite string, and so, need not be A . These are the only observations needed. \square

Now we break the proof of the extended Posner-Robinson theorem into cases. For the moment, imagine we are trying to do a single instance of our goal:

$$[\mathbf{d} \not\leq_h \mathbf{c} \ \& \ (\mathbf{a} \not\leq_h \mathbf{c} \text{ or } \mathbf{d} \not\leq_h \mathbf{b} \sqcup \mathbf{c})] \rightarrow (\exists g)[\mathbf{b} \leq_h \mathbf{a} \sqcup g \ \& \ \mathbf{d} \not\leq_h \mathbf{c} \sqcup g]$$

for some hyperarithmetical degrees $\mathbf{a}, \mathbf{b}, \mathbf{c}$ and \mathbf{d} . Suppose they satisfy the antecedent, then, either $\mathbf{a} \not\leq_h \mathbf{c}$, or, $\mathbf{a} \leq_h \mathbf{c}$ and $\mathbf{d} \not\leq_h \mathbf{b} \sqcup \mathbf{c}$. We call the former the easy case, and the latter the hard case. Pick representatives A, B, C and D from $\mathbf{a}, \mathbf{b}, \mathbf{c}$ and \mathbf{d} respectively, furthermore, if we are in the hard case, ensure that $A \leq_T C$.

3 The easy case

We now proceed with the easy case, for which we use the forcing \mathcal{P} . While constructing our generic, we will need to code B into the join of A and Φ_G . Our coding strategy is to add axioms $\langle n, B(n), \alpha \rangle$ to our generic, with $\alpha \subset A$. To ensure that this is possible, we need to show that we can construct a generic sequence without adding A to the infinite part of any condition, and, such that any axiom $\langle x, y, \sigma \rangle$ applying to A that we add to the finite part satisfies $B(x) = y$.

Additionally, we need to show that a sufficiently generic sequence ensures that $D \not\leq_h C \oplus \Phi_G$. Towards this goal we use the forcing language $\mathcal{L}_{\omega_1, \omega}^r(C)$. As our language can only express reductions from some C -recursive jump of $C \oplus \Phi_G$ we must also show that Φ_G preserves ω_1^C . Thus, we have three goals: coding B , diagonalizing against computing D , and preserving ω_1^C . In this section we work towards the first two goals, leaving the last for Section 5.

Coding B into the join

Definition 3.1 ($((\mathcal{P}, C)$ -essential to $\neg\psi$ over Φ_0). Let Φ_0 be a finite use-monotone Turing functional and

$$\psi = \bigwedge_i (\forall \vec{u}_i) \theta_i(\vec{u}_i)$$

a $\Pi_\alpha^r(C)$ sentence in $\mathcal{L}_{\omega_1, \omega}^r(C)$ for some $\alpha > 0$.

For $\tau = (\tau_1, \dots, \tau_n)$, a sequence of elements of $2^{<\omega}$ all of the same length, we say that τ is (\mathcal{P}, C) -**essential to $\neg\psi$ over Φ_0** , if, for all $p \in \mathcal{P}$, all i , and all \vec{m} of the correct length, if p is a condition such that $p <_{\mathcal{P}} (\Phi_0, \emptyset)$, and $p \Vdash_{\mathcal{P}} \neg\theta_i(\vec{m})$, then, $\Phi_p \setminus \Phi_0$ includes a triple $\langle x, y, \sigma \rangle$ such that σ is compatible with at least one component of τ .

Definition 3.2 ($T_{\mathcal{P}, C}(\Phi_0, \psi, k)$). For each finite use-monotone Turing functional Φ_0 , each $\Pi_\alpha^r(C)$ sentence ψ of our forcing language with $\alpha > 0$, and each natural number k , let $T_{\mathcal{P}, C}(\Phi_0, \psi, k)$ be the set of length k vectors of binary strings all of the same length that are (\mathcal{P}, C) -essential to $\neg\psi$ over Φ_0 .

We drop the (\mathcal{P}, C) prefix or subscript where confusion will not arise (as in the rest of this section, where we have fixed C and are only dealing with \mathcal{P}).

We order $T(\Phi_0, \psi, k)$ by extension on all coordinates. Being essential is closed downward in this order. This endows $T(\Phi_0, \psi, k)$ with the structure of a subtree of the length k vectors of binary sequences of equal length, so $T(\Phi_0, \psi, k)$ is a subtree of a recursively bounded recursive tree.

Lemma 3.3. *Suppose that Φ_0 is a finite use-monotone Turing functional, ψ is a $\Pi_\alpha^r(C)$ sentence in our forcing language with $\alpha > 0$, and k is a natural number.*

- (1) *If there is a size k set \mathbf{X} of reals such that $(\Phi_0, \mathbf{X}) \Vdash \psi(\Phi_G)$, then $T(\Phi_0, \psi, k)$ is infinite.*
- (2) *If $T(\Phi_0, \psi, k)$ is infinite then it has an infinite path Y . Further, each such Y is naturally identified with a size k set $\mathbf{X}(Y)$ of reals such that $(\Phi_0, \mathbf{X}(Y)) \Vdash \psi(\Phi_G)$.*

Proof. For the first claim, let \mathbf{X} be such a set and let (X_1, \dots, X_k) be an enumeration of it, further, let τ_l denote $(X_1 \upharpoonright l, \dots, X_k \upharpoonright l)$. We show that each $\tau_l \in T(\Phi_0, \psi, k)$, proving the tree is infinite.

Suppose $(\forall \vec{u}_i) \theta_i(\vec{u}_i)$ is one of the conjuncts that makes up ψ , \vec{m} is the same length as \vec{u}_i , and $p \leq_{\mathcal{P}} (\Phi_0, \emptyset)$ forces $\neg\theta_i(\vec{m})$, so consequently, $p \Vdash \neg\psi$. As p forces $\neg\psi$ and (Φ_0, \mathbf{X}) forces ψ , these two conditions are incompatible in \mathcal{P} , and as $p \leq_{\mathcal{P}} (\Phi_0, \emptyset)$ there must be an axiom $\langle x, y, \sigma \rangle \in \Phi_p$ which is forbidden from entering any extension of (Φ_0, \mathbf{X}) . But then, σ is an initial segment of some $X_i \in \mathbf{X}$ and so, σ is compatible with the corresponding component of τ_l , showing τ_l is essential.

For the second claim, suppose $T(\Phi_0, \psi, k)$ is infinite. As it is a subtree of a finitely branching tree, König's lemma provides an infinite path Y through it. To each such Y , associate $\mathbf{X}(Y)$, the componentwise union of the coordinates of the members of Y . This will be a size k set of reals. We claim $(\Phi_0, \mathbf{X}(Y))$ forces ψ for each such Y .

Suppose $p < (\Phi_0, \emptyset)$, and there is an i and an \vec{m} such that $p \Vdash \neg\theta_i(\vec{m})$. Then there is an axiom $\langle x, y, \sigma \rangle \in \Phi_p \setminus \Phi_0$ where σ is compatible with some component of each element of Y . In particular, for sufficiently large elements of Y , σ is an initial segment of such a component and so is an initial segment of some $X \in \mathbf{X}(Y)$. So then, p does not extend $(\Phi_0, \mathbf{X}(Y))$. Therefore, no extension of $(\Phi_0, \mathbf{X}(Y))$ forces $\neg\theta_i(\vec{m})$ for any i and \vec{m} and as each extension of $(\Phi_0, \mathbf{X}(Y))$ has a further extension deciding $\theta_i(\vec{m})$, by the definition of forcing, $(\Phi_0, \mathbf{X}(Y)) \Vdash \psi$. \square

Notation. For a set $S \subset \omega$, we say a decision procedure or property of natural numbers is $\Pi_\alpha^0(S)$ for an S -recursive ordinal $\alpha > 0$, if the set of solutions to the procedure or the property, respectively, is co-r.e. in $S^{(\alpha)}$ for $\alpha \geq \omega$, and co-r.e. in $S^{(\alpha-1)}$ for finite α . We say it is $\Delta_\alpha^0(S)$ if it and its complement are both $\Pi_\alpha^0(S)$. For example a $\Pi_1^0(S)$ set is co-r.e. in S but a $\Pi_{\omega+1}^0(S)$ set is co-r.e. in $S^{\omega+1}$.

Lemma 3.4. *For each finite use-monotone Turing functional Φ_0 , each $\Pi_\alpha^r(C)$ sentence ψ with $\alpha > 0$, and each number k , $T(\Phi_0, \psi, k)$ is $\Pi_\alpha^0(C)$ uniformly in Φ_0 , ψ and k .*

Proof. Fix a length k vector τ of binary strings of the same length. Recall that τ is essential to $\neg\psi$ over Φ_0 if for all $p = (\Phi_p, \mathbf{X}_p) <_{\mathcal{P}} (\Phi_0, \emptyset)$, all conjuncts $(\forall \vec{u})\theta_i(\vec{u})$ which make up ψ , and all \vec{m} of the correct length, if $p \Vdash \neg\theta_i(\vec{m})$, then $\Phi_p \setminus \Phi_0$ contains an axiom $\langle x, y, \sigma \rangle$ where σ is compatible with some component of τ .

We can express this, equivalently, as

For all $\Phi_p \leq_{\mathcal{P}} \Phi_0$, all \vec{m} of the correct length, and all conjuncts $(\forall \vec{u})\theta_i(\vec{u})$ which make up ψ , if there exists a set \mathbf{X} such that $(\Phi_p, \mathbf{X}) \Vdash \neg\theta_i(\vec{m})$, then $\Phi_p \setminus \Phi_0$ contains an axiom $\langle x, y, \sigma \rangle$ where σ is compatible with some component of τ .

Or more compactly as

$$\begin{aligned} & (\forall \Phi_p)(\forall \vec{m})(\forall i)[((\Phi_p, \emptyset) \leq_{\mathcal{P}} (\Phi_0, \emptyset) \ \& \ (\exists \mathbf{X})(\Phi_p, \mathbf{X}) \Vdash \neg\theta_i(\vec{m})) \\ & \rightarrow ((\exists \langle x, y, \sigma \rangle \in \Phi_p \setminus \Phi_0) \text{ with } \sigma \text{ compatible with some } \tau \in \tau)]]. \end{aligned}$$

We can also “quantify out” the size of the set \mathbf{X} , as

$$\begin{aligned} & (\forall \Phi_p)(\forall \vec{m})(\forall j)(\forall i)[((\Phi_p, \emptyset) \leq_{\mathcal{P}} (\Phi_0, \emptyset) \ \& \ (\exists \mathbf{X}, |\mathbf{X}| = j)(\Phi_p, \mathbf{X}) \Vdash \neg\theta_i(\vec{m})) \\ & \rightarrow ((\exists \langle x, y, \sigma \rangle \in \Phi_p \setminus \Phi_0) \text{ with } \sigma \text{ compatible with some } \tau \in \tau)]]. \end{aligned}$$

We use both of these formulations in an inductive proof of this lemma, the first in the base case, and the second in the inductive steps. Note that in both cases, the $(\forall i)$ is a quantifier over a C -r.e. set: the conjuncts making up ψ . This is still equivalent to a single universal quantifier.

Suppose $\alpha = 1$. In this case, for each i and \vec{m} of the correct length $\neg\theta_i(\vec{m})$ is a quantifier-free sentence. So, by Lemma 2.6, deciding whether there is an \mathbf{X} such that $(\Phi_p, \mathbf{X}) \Vdash \neg\theta_i(\vec{m})$ is uniformly C -recursive in Φ_p, θ_i and \vec{m} . Also, checking whether $\Phi_p \leq_{\mathcal{P}} \Phi_0$ and whether there is an axiom $\langle x, y, \sigma \rangle \in \Phi_p \setminus \Phi_0$ compatible with some $\tau \in \tau$ is uniformly recursive in Φ_0, Φ_p and τ .

Consequently, in the case $\alpha = 1$, τ being essential to $\neg\theta$ over Φ_0 can be expressed as a property with two universal natural number quantifiers, then a universal quantifier over a C -r.e. set, then a matrix which is C -recursive uniformly in all parameters. Such an expression is co-r.e. in C , or $\Pi_1^0(C)$ as required.

If $\alpha > 1$ is a successor, we can assume all its conjuncts are $\Sigma_{\alpha-1}^r(C)$ by Ash and Knight. By Lemma 3.3, there being an \mathbf{X} of size j such that $(\Phi_p, \mathbf{X}) \Vdash \neg\theta_i(\vec{m})$ is equivalent to $T(\Phi_p, \neg\theta_i(\vec{m}), j)$ being infinite. But $T(\Phi_p, \neg\theta_i(\vec{m}), j)$ is, by induction, a $\Pi_\alpha^0(C)$ subtree of

a recursively bounded recursive tree. Hence, it being infinite is also a $\Pi_\alpha^0(C)$ fact, given uniformly in Φ_p, j, θ_i and \vec{m} . (This bound follows because the tree being finite is equivalent to it being of bounded height and our trees are subtrees of recursive trees. Hence, saying our trees is finite is equivalent to saying there is a level of a recursive tree, from which our tree is disjoint i.e. $(\exists n)[\text{for all nodes } x \text{ in the recursive tree } T \text{ at level } n, x \text{ is not in our tree}]$. If our tree is $\Pi_\alpha^0(C)$ for some C this formula is equivalent to a $\Sigma_\alpha^0(C)$ formula, and so its negation is $\Pi_\alpha^0(C)$ as required.)

Thus, for successor α , τ being essential to $\neg\psi$ over Φ_0 is equivalent to a property which we can express with a block of universal natural number quantifiers, then a universal quantifier over a C -r.e. set, then a $\Sigma_{\alpha-1}^0(C)$ matrix uniformly in the parameters. This renders the property $\Pi_\alpha^0(C)$ as required.

For the limit case the argument is similar. The same formulation of the definition of τ being essential works as for the successor case. However, here we have the full power of $C^{(\alpha)}$ which can decide whether recursively branching $C^{(\beta)}$ -recursive trees are infinite, for $\beta < \alpha$, uniformly in an index for the tree, so, $C^{(\alpha)}$ can uniformly decide the matrix. The prefix is a block of universal quantifiers, then a universal quantifier over a C -r.e. set, with uniformly $C^{(\alpha)}$ -recursive matrix, so τ being essential is uniformly co-r.e. in $C^{(\alpha)}$, or $\Pi_\alpha^0(C)$ as required. \square

Corollary 3.5. *Suppose that $S \subseteq \omega$ is not $\Delta_\alpha^0(C)$. Let Φ_0 be a finite use-monotone Turing functional, ψ a $\Pi_\alpha^r(C)$ sentence with $\alpha > 0$, and $k \geq 1$. If there is a size k set \mathbf{X} of reals such that $(\Phi_0, \mathbf{X}) \models \psi$, then there is such a set not containing S . Moreover, we can find such an \mathbf{X} all of whose members are recursive in $C^{(\alpha+1)}$ uniformly in ψ, k and $S \oplus C^{(\alpha+1)}$.*

Proof. Suppose there is a size k set \mathbf{X} such that $(\Phi_0, \mathbf{X}) \models \psi$. By Lemmas 3.3 and 3.4 the tree $T(\Phi_0, \psi, k)$ is a $\Pi_\alpha^0(C)$ subtree of a recursively bounded recursive tree which has an infinite path. The paths form a nonempty $\Pi_\alpha^0(C)$ class and so there is a path Y in which S is not recursive. Then, certainly S is not a member of $\mathbf{X}(Y)$. Further, Lemma 3.3 implies $(\Phi_0, \mathbf{X}(Y)) \models \psi$. This completes the proof of the first claim.

For the second claim, we need to find an infinite path in $T(\Phi_0, \psi, k)$ on which S does not appear, recursively in $C^{(\alpha+1)} \oplus S$. We need $C^{(\alpha+1)}$ to decide which members of $T(\Phi_0, \psi, k)$ have an infinite part of $T(\Phi_0, \psi, k)$ above it, and S to decide whether a given node could possibly have an extension on which S appears. Consequently, there is an algorithm, recursive in $S \oplus C^{(\alpha+1)}$, solving this problem which, searches through the tree of length k sequences of binary strings of the same length, looking for one which disagrees with S and has an infinite part of $T(\Phi_0, \psi, k)$ above it. Once one is found, the algorithm switches to searching for any path in $T(\Phi_0, \psi, k)$ through this node. Note the uniformity in ψ and k follows from the uniformity in the construction of $T(\Phi_0, \psi, k)$.

Of course, once we have found a node each of the components of which are not compatible with S , yet has a path through it, we never use S again in the construction. Hence, we could externally hard code to only search for paths in $T(\Phi_0, \psi, k)$ through some fixed node, and any such path Y corresponds to reals $\mathbf{X}(Y)$ which are recursive in $C^{(\alpha+1)}$ (as $C^{(\alpha+1)}$ is constructing such a Y), hence, the members of the \mathbf{X} we build are $C^{(\alpha+1)}$ -recursive although we lose the uniformity by externally hard-coding in some information. \square

Corollary 3.6. *Suppose that $\alpha > 0$, S is not $\Delta_\alpha^0(C)$ and ψ is a $\Pi_\alpha^r(C)$ sentence. For any condition $p = (\Phi_p, \mathbf{X}_p)$ with $S \notin \mathbf{X}_p$, there is a stronger q which we can find uniformly in Φ_p and $S \oplus C^{(\alpha+1)} \oplus \mathbf{X}_p$ such that*

- (1) $S \notin \mathbf{X}_q$ and each $X \in \mathbf{X}_q \setminus \mathbf{X}_p$ is recursive in $C^{(\alpha+1)}$.
- (2) For all x , if $\Phi_q(S)(x)$ is defined, then $\Phi_p(S)(x)$ is defined. That is, q does not add any new computations which apply to S .
- (3) Either $q \Vdash \psi$ or there is a conjunct θ_i and an \vec{m} such that $q \Vdash \neg\theta_i(\vec{m})$ (and we can tell which formula we have forced).

Proof. Fix a condition $p = (\Phi_p, \mathbf{X}_p)$, and let $k = |\mathbf{X}_p|$. We would like to know whether $T(\Phi_0, \psi, k+1)$ is infinite. It is a $\Pi_\alpha^0(C)$ subtree of a recursively bounded recursive tree, and hence, this is a $\Pi_\alpha^0(C)$ fact. If $T(\Phi_0, \psi, k+1)$ is infinite, then Corollary 3.5 provides a path Y through it, not containing S , each member of which is recursive in $S \oplus C^{(\alpha+1)}$. Then, $(\Phi_p, \mathbf{X}_p \cup \mathbf{X}(Y)) \leq_{\mathcal{P}} (\Phi_p, \mathbf{X})$, $(\Phi_p, \mathbf{X}_p \cup \mathbf{X}(Y)) \Vdash \psi$, and, clearly, (2) is satisfied, as we haven't changed the finite part.

If $T(\Phi_0, \psi, k+1)$ is not infinite, then (enumerating $\mathbf{X} = \{X_1, \dots, X_k\}$) (X_1, \dots, X_k, S) does not provide a path through it. Consequently, for some l , $(X_1 \upharpoonright l, \dots, X_n \upharpoonright l, S \upharpoonright l)$ is not essential to $\neg\psi$ over Φ_p , meaning there is a conjunct θ_i , numerals \vec{m} and a condition $q = (\Phi_q, \mathbf{X}_q)$ extending (Φ_p, \emptyset) such that $q \Vdash_{\mathcal{P}} \neg\theta_i(\vec{m})$ yet, every new axiom $\langle x, y, \sigma \rangle \in \Phi_q \setminus \Phi_p$ is incompatible with each coordinate of $(X_1 \upharpoonright l, \dots, X_n \upharpoonright l, S \upharpoonright l)$. In particular, there are no new axioms applying to S , nor to any $X_i \in \mathbf{X}$, and by Lemma 3.3 there is a j such that $T(\Phi_q, \neg\theta_i(\vec{m}), j)$ is infinite.

We can find which $\Phi_q < \Phi_p$ add neither any new computations applying to S , nor to any member of \mathbf{X}_p , recursively in $S \oplus \mathbf{X}_p$, and then, whether $T(\Phi_q, \neg\theta_i(\vec{m}), j)$ is infinite recursively in $C^{(\alpha)}$. Consequently, we can find such a $\Phi_q < \Phi_p, \theta_i, \vec{m}$ and j recursively in $S \oplus C^{(\alpha+1)} \oplus \mathbf{X}_p$, (we add the extra jump because for each $\Phi_q < \Phi_p$ we ask whether there exists a θ_i, \vec{m} and j such that $T(\Phi_p, \neg\theta_i(\vec{m}), j)$ is infinite). Once we have found such a $\Phi_q, \theta_i, \vec{m}$ and j , applying Corollary 3.6 again, there is a size j set \mathbf{X} with $S \notin \mathbf{X}$ such that every $X \in \mathbf{X}$ is uniformly recursive in $S \oplus C^{(\alpha+1)}$ and $(\Phi_q, \mathbf{X}) \Vdash \neg\theta_i(\vec{m})$, therefore, $(\Phi_q, \mathbf{X} \cup \mathbf{X}_p)$ is the condition we want. \square

As A is not $\Delta_\alpha^0(C)$ for any C -recursive α , we can repeatedly apply Corollary 3.6 to construct an $\mathcal{L}_{\omega_1, \omega}^r(C)$ -generic sequence in \mathcal{P} , without adding A to the infinite part of any condition, and without adding any axioms applying to A . Hence, we can dovetail this construction with one which adds axioms $\langle n, B(n), \alpha \rangle$ for some sufficiently long $\alpha \subset A$ which effectively codes B into the join of A and Φ_G . Now we proceed with showing that we can diagonalize against $C \oplus \Phi_G$ computing D .

Diagonalizing against D

Notation. For a real S , we say $\langle e, \alpha \rangle$ is an S -hyperarithmetic reduction, if e is an index and α is an S -recursive ordinal. We write $\langle e, \alpha \rangle^S$ for the partial function $\{e\}^{S^{(\alpha)}}$.

Definition 3.7 ((\mathcal{P}, C)-Essential to splits). Let Φ_0 be a finite use-monotone Turing functional and $\langle e, \alpha \rangle$ a C -hyperarithmetical reduction. For each τ , a finite sequence of binary strings of the same length, we say τ is (\mathcal{P}, C)-**essential to $\langle e, \alpha \rangle$ -splits below Φ_0** if whenever $p, q <_{\mathcal{P}} (\Phi_0, \emptyset)$ form an $\langle e, \alpha \rangle$ -split (i.e. there is an x such that $p \Vdash \langle e, \alpha \rangle^{C \oplus \Phi_G}(x) \downarrow = k_1, q \Vdash \langle e, \alpha \rangle^{C \oplus \Phi_G}(x) \downarrow = k_2$ and $k_1 \neq k_2$) there is some new axiom $\langle x, y, \sigma \rangle \in \Phi_p \cup \Phi_q$ but not in Φ_0 such that σ is compatible with some component of τ .

We let $U_{\mathcal{P}, C}(\Phi_0, \langle e, \alpha \rangle, k)$ be the set of τ of length k which are (\mathcal{P}, C)-essential to $\langle e, \alpha \rangle$ -splits below Φ_0 , and consider this as a tree as before. Also, as before, we drop the \mathcal{P} and C when confusion won't arise.

Lemma 3.8. *Let Φ_0 be a Turing functional, $\langle e, \alpha \rangle$ a C -hyperarithmetical reduction, and k a natural number.*

- *If \mathbf{X} is a size k set of reals such that no pair $p, q <_{\mathcal{P}} (\Phi_0, \mathbf{X})$ form an $\langle e, \alpha \rangle$ -split, then $U(\Phi_0, \langle e, \alpha \rangle, k)$ is infinite.*
- *If $U(\Phi_0, \langle e, \alpha \rangle, k)$ is infinite then it has an infinite path, and each such path Y is identified with a size k set $\mathbf{X}(Y)$ such that $(\Phi_0, \mathbf{X}(Y))$ has no $\langle e, \alpha \rangle$ -splits below it.*

Proof. Let $\tau_l = (X_1 \upharpoonright l, \dots, X_k \upharpoonright l)$ where X_1, \dots, X_k enumerates \mathbf{X} . Suppose $p, q <_{\mathcal{P}} (\Phi_0, \emptyset)$ form an $\langle e, \alpha \rangle$ -split. As (Φ_0, \mathbf{X}) has no splits below it at least one of p and q are not compatible with (Φ_0, \mathbf{X}) . Suppose it is p which is incompatible. As $p <_{\mathcal{P}} (\Phi_0, \emptyset)$, p must have a new axiom $\langle x, y, \sigma \rangle$ which some X_i forbids. Hence, $\sigma \subset X_i$ and, therefore, σ is compatible with the i th component of τ_l . Hence, every τ_l is essential to $\langle e, \alpha \rangle$ -splits, and so $U(\Phi_0, \langle e, \alpha \rangle, k)$ is infinite.

For the second claim, suppose $U(\Phi_0, \langle e, \alpha \rangle, k)$ is infinite. Then, Konig's lemma provides us a path Y , and $\mathbf{X}(Y)$ is the componentwise union of Y . Suppose $p, q \leq_{\mathcal{P}} (\Phi_0, \mathbf{X}(Y))$ form an $\langle e, \alpha \rangle$ -split below $(\Phi_0, \mathbf{X}(Y))$, then they also form a split below (Φ_0, \emptyset) . As the elements of Y are essential, there is some new axiom $\langle x, y, \sigma \rangle$ in (WLOG) Φ_p , but not in Φ_0 compatible with some component of each element of Y . Hence, σ is an initial segment of some component of $\mathbf{X}(Y)$, and so $p \not\leq_{\mathcal{P}} (\Phi_0, \mathbf{X}(Y))$, a contradiction. \square

Lemma 3.9. *If Φ_0 is a Turing functional, $\langle e, \alpha \rangle$ a C -hyperarithmetical reduction, and k a natural number then, $U(\Phi_0, \langle e, \alpha \rangle, k)$ is $\Pi_{\alpha+3}^0(C)$ uniformly in Φ_0 , e , and k .*

Proof. Fix $\Phi_0, \langle e, \alpha \rangle, k$, and τ of length k . Then $\tau \in U(\Phi_0, \langle e, \alpha \rangle, k)$ if and only if τ is essential to splits below Φ_0 , which is equivalent to:

$$\begin{aligned}
& (\forall p, q <_{\mathcal{P}} (\Phi_0, \emptyset)) (\forall m, m_1, m_2) [(p \Vdash \langle e, \alpha \rangle^{(C \oplus \Phi_G)}(m) \downarrow = m_1 \\
& \quad \wedge \quad q \Vdash \langle e, \alpha \rangle^{(C \oplus \Phi_G)}(m) \downarrow = m_2 \\
& \quad \wedge \quad m_1 \neq m_2) \\
& \quad \rightarrow (\exists \langle x, y, \sigma \rangle) \text{ such that } \langle x, y, \sigma \rangle \in (\Phi_p \cup \Phi_q) \setminus \Phi_0 \\
& \quad \text{yet } \sigma \text{ compatible with some } \tau \in \tau)]
\end{aligned}$$

Using a similar trick as in Lemma 3.4, we rewrite this as

$$\begin{aligned}
& (\forall \Phi_p, \Phi_q \leq_P \Phi_0) (\forall m, m_1, m_2) [((\exists \mathbf{X}_p)(\Phi_p, \mathbf{X}_p) \Vdash \langle e, \alpha \rangle^{(C \oplus \Phi_G)}(m) \downarrow = m_1 \\
& \quad \wedge (\exists \mathbf{X}_q)(\Phi_q, \mathbf{X}_q) \Vdash \langle e, \alpha \rangle^{(C \oplus \Phi_G)}(m) \downarrow = m_2 \\
& \quad \wedge m_1 \neq m_2) \\
& \quad \rightarrow (\exists \langle x, y, \sigma \rangle) \text{ such that } \langle x, y, \sigma \rangle \in (\Phi_p \cup \Phi_q) \setminus \Phi_0 \\
& \quad \text{yet } \sigma \text{ compatible with some } \tau \in \boldsymbol{\tau}]
\end{aligned}$$

But, by Lemma 3.3, we can replace

$$(\exists \mathbf{X}_p) [(\Phi_p, \mathbf{X}_p) \Vdash \langle e, \alpha \rangle^{(C \oplus \Phi_G)}(m) \downarrow = m_1]$$

with

$$(\exists j) [T(\Phi_p, \langle e, \alpha \rangle^{C \oplus \Phi_G}(m) \downarrow = m_1, j) \text{ is infinite}]$$

and make similar replacements for Φ_q and \mathbf{X}_q .

Now, “ $\langle e, \alpha \rangle^{(C \oplus \Phi_G)}(m) \downarrow = m_1$ ” can be expressed as a $\Sigma_{\alpha+1}^r(C)$ sentence in our forcing language, uniformly in m, m_1, e, α . So then, we would like to know whether a $\Pi_{\alpha+2}^0(C)$ subtree of a recursively bounded recursive tree (have to go up one level one to get a Π^r sentence) is infinite, which is a $\Pi_{\alpha+2}^0(C)$ condition, uniformly in j, m, m_1, Φ_p . Hence, $\boldsymbol{\tau}$ being essential to $\langle e, \alpha \rangle$ -splits below Φ_0 is equivalent to a sentence with an initial block of universals, with a conditional matrix, the antecedent of which is $\Pi_{\alpha+2}^0(C)$ and the consequent uniformly recursive. Hence, $\boldsymbol{\tau}$ being essential to $\langle e, \alpha \rangle$ -splits below Φ_0 is a $\Pi_{\alpha+3}(C)$ property. Uniformity is clear from the analysis. \square

Corollary 3.10. *For each finite use-monotone Turing functional Φ_0 , each C -hyperarithmetic reduction $\langle e, \alpha \rangle$, and each natural number k , if S is not $\Delta_{\alpha+3}^0(C)$ and if there is a size k set \mathbf{X} such that (Φ_0, \mathbf{X}) has no $\langle e, \alpha \rangle$ -splits below it, then there is such an \mathbf{X} not containing S . Further, we can find such an \mathbf{X} all of whose members are recursive in $C^{(\alpha+4)}$ indeed, we can construct \mathbf{X} uniformly in e, k and $S \oplus C^{(\alpha+4)}$*

Proof. Suppose there is a size k set \mathbf{X} such that (Φ_0, \mathbf{X}) has no $\langle e, \alpha \rangle$ -splits. Then, by Lemma 3.8, $U(\Phi_0, \langle e, \alpha \rangle, k)$ is infinite, and by Lemma 3.9 it is uniformly $\Pi_{\alpha+3}^0(C)$. By the same argument as for Corollary 3.5, this tree has a path, Y , recursive in $C^{(\alpha+4)}$, and yet, S is not recursive in the members of the path, and consequently, S is not on the path. Furthermore, by Lemma 3.8, the reals $\mathbf{X}(Y)$ serve as the second coordinate of a condition $(\Phi_0, \mathbf{X}(Y))$, which has no $\langle e, \alpha \rangle$ -splits below it.

Now, for the uniform construction, we need to build a path through $U(\Phi_0, \langle e, \alpha \rangle, k)$, a $\Pi_{\alpha+3}^0(C)$ recursively bounded tree, and we need to make sure S is not on the path. $C^{(\alpha+4)}$ suffices to decide which members of the tree have an infinite part of the tree above them. Additionally S can check whether a given node has coordinates all of which disagree with S , and as $U(\Phi_0, \langle e, \alpha \rangle, k)$ is given uniformly in the mentioned parameters, we have the desired uniformity. \square

Corollary 3.11. *Suppose $p \in \mathcal{P}$ is a forcing condition such that $A \notin \mathbf{X}_p$, and $\langle e, \alpha \rangle$ a C -hyperarithmetic reduction. Then, there is an extension of p which adds no new axioms applying to A and does not have A in the infinite part, which diagonalizes against $\langle e, \alpha \rangle^{C \oplus \Phi_G} = D$ i.e. either forces $\langle e, \alpha \rangle^{C \oplus \Phi_G}$ is not total or there is an x such that it forces this computation on x to be convergent, yet not equal to $D(x)$.*

Proof. Fix p . The expression “ $\langle e, \alpha \rangle^{C \oplus \Phi_G}$ is total” can be rendered in our forcing language as a $\Pi_{\alpha+2}^r(C)$ sentence. Then, by Corollary 3.6, we can find a stronger condition q neither adding A to the infinite part nor any new axioms applying to A to the finite part, deciding this sentence. If it forces it to be false (i.e. the reduction is not total), q is the extension we want.

Otherwise suppose q forces totality. If q has an $\langle e, \alpha \rangle^{C \oplus \Phi_G}$ split, then there is an $r \leq_p q$ and an x such that $r \Vdash \langle e, \alpha \rangle^{C \oplus \Phi_G}(x) \downarrow \neq D(x)$. As there is an $r \leq_p q$ forcing this sentence, then by Corollary 3.5 there is such an r which neither adds A to the infinite part, nor adds any new computations applying to A to the finite part. Such an r is the extension we want.

Now suppose q has no such splits. Then, $\langle e, \alpha \rangle^{C \oplus \Phi_G}$ is determined by q . We claim we can hyperarithmetically in C determine what set q determines this to be, and so compute $\langle e, \alpha \rangle^{C \oplus \Phi_G}$ (as determined by q) hyperarithmetically in C , which implies it is not D .

To see this, recall that as $q = (\Phi_q, \mathbf{X}_q)$ has no splits, then there is a finite set of reals \mathbf{X} (still not containing A) all of whose members are recursive in $C^{(\alpha+4)}$ such that $q' = (\Phi_q, \mathbf{X})$ also has no splits, by Corollary 3.10. Note q' and q are compatible.

Now, for fixed x and each y and k , search for an Φ_r such that $(\Phi_r, \mathbf{X}) \leq_p (\Phi_q, \mathbf{X})$, and $T(\Phi_r, \langle e, \alpha \rangle^{C \oplus \Phi_G}(x) \downarrow = y, k)$ is infinite. Checking the first condition is recursive in $C^{(\alpha+4)}$, and so too is the second, as, the tree is $\Pi_{\alpha+2}^0(C)$, uniformly in $\Phi_r, e, \alpha, x, y, k$, hence, this search is $C^{(\alpha+4)}$ -recursive. Once we find such a y, k we can output y as the value of $\langle e, \alpha \rangle^{C \oplus \Phi_G}(x)$. As q' and q are compatible and q' has no splits, y must be the value that q decides for $\langle e, \alpha \rangle^{C \oplus \Phi_G}(x)$. We know such Φ_r, y , and k exist, because if they didn't, then we could force nonconvergence of the computation at x , contrary to hypothesis.

Thus, recursively in $C^{(\alpha+3)}$ we can compute what q forces $\langle e, \alpha \rangle^{C \oplus \Phi_G}$ to be, and so, as D is not hyperarithmetic in C , this computation does not compute D . \square

So, we have shown that we can construct a generic sequence for the easy case, while still effecting the coding. We will postpone the proof of preservation of ω_1^C until we have proved results analogous to the above for the hard case.

4 The hard case

In the previous section, we proved results which we will apply to A and C , provided A is not hyperarithmetic in C i.e. if we are in the easy case. In the hard case, these lemmas no longer apply. This is the motivation for the forcing $\mathcal{Q}_{A,B}$ (which we abbreviate as \mathcal{Q} in this section), where we build into the notion of forcing the lemmas that we can't push through for \mathcal{P} . Recall that, in the hard case, we choose representatives such that $A \leq_T C$. In this section when we say a condition forces a sentence, we mean $\Vdash_{\mathcal{Q}}$.

Note that if we are successful in our goal of coding B into the join of A and Φ_G , then in the hard case we will have

$$C \oplus \Phi_G \geq_T A \oplus \Phi_G \geq_T B$$

and, consequently, $C \oplus \Phi_G \geq_T B \oplus C$. Thus, we will attempt to preserve the fact that $B \oplus C \not\leq_h D$ in the hard case. This requires us to use the language $\mathcal{L}_{\omega_1, \omega}^r(B \oplus C)$ as our language of forcing for \mathcal{Q} .

Definition 4.1 ($(\mathcal{Q}, B \oplus C)$ -essential to $\neg\psi$ over Φ_0). If Φ_0 is a finite use-monotone Turing functional which partially computes B given A , ψ is a $\Pi_\alpha^r(B \oplus C)$ sentence in $\mathcal{L}_{\omega_1, \omega}^r(B \oplus C)$, and τ a finite tuple of binary strings, all of the same length, we say τ is $(\mathcal{Q}, B \oplus C)$ -essential to $\neg\psi$ over Φ_0 , if, whenever $q = (\Phi_q, \mathbf{X}_q) \in \mathcal{Q}$ properly extends (Φ_0, \emptyset) and $q \Vdash_{\mathcal{Q}} \neg\theta_i(\vec{m})$ for one of the conjuncts $(\forall \vec{u})\theta_i(\vec{u})$ making up ψ and some tuple \vec{m} of the correct length, $\Phi_q \setminus \Phi_0$ contains an axiom $\langle x, y, \sigma \rangle$ such that σ is compatible with some $\tau \in \tau$.

We let $T_{\mathcal{Q}, B \oplus C}(\Phi_0, \psi, k)$ be the tree of essential tuples of length k , as before. We will abuse notation and omit the subscripts of $T(\Phi_0, \psi, k)$. Again, $T(\Phi_0, \psi, k)$ is a subtree of a recursively bounded recursive tree. Note that $T(\Phi_0, \psi, k)$ could have branches which contain A as the limit of one of the coordinates, meaning we can't use that branch as the set of reals for a \mathcal{Q} condition. Given a path Y in the tree $T(\Phi_0, \psi, k)$ (or any of our trees) and a real S we say S **does not appear on** Y if S is not the limit of any of the coordinates of Y .

Lemma 4.2. *Suppose that Φ_0 is a finite use-monotone Turing functional which partially computes B on input A , ψ is a $\Pi_\alpha^r(B \oplus C)$ sentence with $\alpha > 0$ and k is a natural number.*

- (1) *If there is a size k set \mathbf{X} of reals such that $(\Phi_0, \mathbf{X}) \in \mathcal{Q}$ and forces $\psi(\Phi_G)$, then $T(\Phi_0, \psi, k)$ has an infinite path on which A does not appear.*
- (2) *If $T(\Phi_0, \psi, k)$ has an infinite path on which A does not appear, then each such path Y is naturally identified with a size k set $\mathbf{X}(Y)$ (not containing A) of reals such that $(\Phi_0, \mathbf{X}(Y)) \in \mathcal{Q}$ and forces $\psi(\Phi_G)$.*

Proof. The proof is similar to that of Lemma 3.3. For the first claim, note that, for each l , $\tau_l = (X_1 \upharpoonright l, \dots, X_k \upharpoonright l)$ is essential (where (X_1, \dots, X_k) enumerates such an \mathbf{X}), as, any $q \in \mathcal{Q}$ extending (Φ_0, \emptyset) and forcing $\neg\theta_i(\vec{m})$ for some conjunct $(\forall \vec{u})\theta_i(\vec{u})$ making up ψ and some \vec{m} , must be incompatible with (Φ_0, \mathbf{X}) . Consequently, there is a new axiom in Φ_q applying to some member of \mathbf{X} and so compatible with the corresponding component of τ_l . Clearly, A does not appear on the path because $(\Phi_0, \mathbf{X}) \in \mathcal{Q}$.

For the second claim, for any path Y in $T(\Phi_0, \psi, k)$ on which A does not appear, $\mathbf{X}(Y)$ - the coordinatewise union of Y - may serve as the infinite part of a \mathcal{Q} condition. Then, as in Lemma 3.3, if $q <_{\mathcal{Q}} (\Phi_0, \emptyset)$ forces $\neg\theta_i(\vec{m})$ for a θ_i and an \vec{m} as above, then q must contain a new axiom compatible with some coordinate of each element of Y , and so, must apply to some $X \in \mathbf{X}(Y)$. Consequently, such a q does not extend (Φ_0, \mathbf{X}) and so $(\Phi_0, \mathbf{X}) \Vdash_{\mathcal{Q}} \psi$. \square

Lemma 4.3. *For each finite use-monotone Turing functional Φ_0 that correctly computes B on input A , each $\Pi_\alpha^r(B \oplus C)$ sentence ψ in $\mathcal{L}_{\omega_1, \omega}^r(B \oplus C)$ with $\alpha > 0$, and each number k , $T(\Phi_0, \psi, k)$ is $\Pi_\alpha^0(B \oplus C)$ uniformly in Φ_0 , ψ , and k .*

Proof. We use the same strategy as for Lemma 3.4. Fix τ of length k , Φ_0 and ψ . Then, τ being essential to $\neg\psi$ over Φ_0 is equivalent to both of the following

$$\begin{aligned} & (\forall \Phi_q)(\forall \vec{m})(\forall i)[((\Phi_q, \emptyset) \leq_Q (\Phi_0, \emptyset) \ \& \ (\exists \mathbf{X})(\Phi_q, \mathbf{X}) \Vdash_Q \neg\theta_i(\vec{m})) \\ & \rightarrow (\exists \langle x, y, \sigma \rangle) \text{ such that } \langle x, y, \sigma \rangle \in \Phi_q \setminus \Phi_0 \\ & \text{yet } \sigma \text{ is compatible with some } \tau \in \tau], \end{aligned}$$

and

$$\begin{aligned} & (\forall \Phi_q)(\forall \vec{m})(\forall j)(\forall i)[((\Phi_q, \emptyset) \leq_Q (\Phi_0, \emptyset) \ \& \ (\exists \mathbf{X}, |\mathbf{X}| = j)(\Phi_q, \mathbf{X}) \Vdash_Q \neg\theta_i(\vec{m})) \\ & \rightarrow (\exists \langle x, y, \sigma \rangle) \text{ such that } \langle x, y, \sigma \rangle \in \Phi_q \setminus \Phi_0 \\ & \text{yet } \sigma \text{ is compatible with some } \tau \in \tau]. \end{aligned}$$

(Again the $(\forall i)$ is a universal quantifier over a $(B \oplus C)$ -r.e. set.)

Now suppose $\alpha = 1$, then by Lemma 2.6, the antecedent in the first formulation is $A \oplus B \oplus C$ -recursive uniformly in Φ_0, Φ_q, θ_i and \vec{m} . As $C \geq_T A$, we can ignore A , and then τ being essential can be rendered as a property with two universal natural number quantifiers, then a universal quantifier over a $(B \oplus C)$ -r.e. set, then a $B \oplus C$ -recursive matrix, uniformly in the parameters. Hence, this is a $\Pi_1^0(B \oplus C)$ property.

For the inductive steps, firstly, suppose $\alpha > 1$ is a successor. We now consider the second equivalent definition of τ being essential. By Lemma 4.2, “ $(\exists \mathbf{X}, |\mathbf{X}| = j)(\Phi_q, \mathbf{X}) \Vdash \neg\theta_i(\vec{m})$ ” is equivalent to $T(\Phi_q, \neg\theta_i(\vec{m}), j)$ having a path on which A does not appear. We may render this as: Does there exist a length l so that for every length $l' > l$ there exists a node on the tree at level l' each coordinate of which is not an initial segment of A ? Because $T(\Phi_q, \neg\theta_i(\vec{m}), j)$ is a subtree of a recursively bounded recursive tree, the second existential quantifier is actually a bounded existential quantifier. As, by induction, $T(\Phi_q, \neg\theta_i(\vec{m}), j)$ is $\Pi_{\alpha-1}^0(B \oplus C)$ and $A \leq_T C$ so the tree having a path on which A does not appear is $\Sigma_\alpha^0(B \oplus C)$.

As this is the antecedent of a conditional, we flip to $\Pi_\alpha^0(B \oplus C)$ and then our property is expressed via an initial block of universal quantifiers, and then a $\Pi_\alpha^0(B \oplus C)$ matrix, for $\Pi_\alpha^0(B \oplus C)$ as required.

Finally, suppose $\alpha > 1$ is a limit. We use the second formulation of τ being essential. Now, by induction, $(B \oplus C)^{(\alpha)}$ can (recursively) decide whether $T(\Phi_p, \neg\theta_i(\vec{m}), j)$ has an infinite path on which A does not appear, for any conjunct $(\forall \vec{u})\theta_i(\vec{u})$ of ψ . Consequently, when α is a limit, the matrix of the second formulation for τ being essential is recursive in $(B \oplus C)^{(\alpha)}$, and the prefix is a block of universals for $\Pi_\alpha^0(B \oplus C)$ as required. \square

Corollary 4.4. *Let Φ_0 be a finite use-monotone Turing functional, ψ a $\Pi_\alpha^r(B \oplus C)$ sentence, and k a natural number. If there is a size k set \mathbf{X} such that $A \notin X$ and $(\Phi_0, \mathbf{X}) \Vdash_Q \psi$, then we can - recursively in $(B \oplus C)^{(\alpha+1)}$ - find such an \mathbf{X} , uniformly in ψ, k (of course, such a set does not contain A).*

Proof. If there is such a set \mathbf{X} , then by Lemma 4.2, $T(\Phi_0, \psi, k)$ has an infinite path on which A does not appear. By Lemma 4.3, $T(\Phi_0, \psi, k)$ is a $\Pi_\alpha^0(B \oplus C)$ subtree of a recursively bounded recursive tree. Consequently, $(B \oplus C)^{(\alpha+1)}$ can decide membership in $T(\Phi_0, \psi, k)$, can decide whether the tree is infinite above any given node, and can decide whether any node is incompatible with A . Thus, $(B \oplus C)^{(\alpha+1)}$ can construct a path Y in $T(\Phi_0, \psi, k)$ on which A does not appear, and then, $(\Phi_0, \mathbf{X}(Y)) \Vdash_{\mathcal{Q}} \psi$. \square

Corollary 4.5. *Let ψ be a $\Pi_\alpha^r(B \oplus C)$ sentence and k a natural number. For any condition $q \in \mathcal{Q}$ there is a stronger $r \leq_{\mathcal{Q}} q$ which we can find uniformly in Φ_q and $(B \oplus C)^{(\alpha+2)} \oplus \mathbf{X}_q$ such that every new $X \in \mathbf{X}_r$ is recursive in $(B \oplus C)^{(\alpha+1)}$, and r decides ψ (and we know whether $r \Vdash_{\mathcal{Q}} \psi$ or $r \Vdash_{\mathcal{Q}} \neg\psi$).*

Proof. Fix a $q \in \mathcal{Q}$ and enumerate \mathbf{X}_q as X_1, \dots, X_k . There are three cases to consider: Either $T(\Phi_q, \psi, k)$ has an infinite path on which A does not appear, or $T(\Phi_q, \psi, k)$ is finite, or $T(\Phi_q, \psi, k)$ is infinite (and so has a path) but every such path contains A .

$T(\Phi_q, \psi, k)$ is a $\Pi_\alpha^0(B \oplus C)$ subtree of a recursively bounded recursive tree by given uniformly in the parameters, by Lemma 4.3. Hence, the question of $T(\Phi_q, \psi, k)$ being infinite is uniformly $\Pi_\alpha^0(B \oplus C)$ and so is decidable in $(B \oplus C)^{(\alpha+1)}$. Furthermore, if it is infinite, $(B \oplus C)^{(\alpha+2)}$ can decide whether there is a path on which A does not appear, because A is recursive in C . Therefore, $(B \oplus C)^{(\alpha+2)}$ can uniformly determine which case we are in.

Case 1: If we find $T(\Phi_q, \psi, k)$ has an infinite path on which A does not appear $(B \oplus C)^{(\alpha+1)}$ suffices to construct such a path Y as in Corollary 4.4. By Lemma 4.2 $(\Phi_q, \mathbf{X}(Y)) \Vdash_{\mathcal{Q}} \psi$ and so $r = (\Phi_q, \mathbf{X}_q \cup \mathbf{X}(Y))$ suffices and was found uniformly.

Case 2: In this case $T(\Phi_q, \psi, k)$ is finite and so (X_1, \dots, X_k) does not form a path through it. Therefore, for some l , $\tau_l = (X_1 \upharpoonright l, \dots, X_k \upharpoonright l)$ is not essential to $\neg\psi$ over Φ_q . Then there is some $p <_{\mathcal{Q}} (\Phi_q, \emptyset)$, some conjunct $\theta_i(\vec{u})$ of ψ and an \vec{m} of the correct length such that, $p \Vdash_{\mathcal{Q}} \neg\theta_i(\vec{m})$ yet no $\langle x, y, \sigma \rangle \in \Phi_p \setminus \Phi_q$ is compatible with any component of τ_l . As $p \Vdash \neg\theta_i(\vec{m})$ there is a j such that $T(\Phi_p, \neg\theta_i(\vec{m}), j)$ has an infinite path on which A does not appear by Lemma 4.2.

$(B \oplus C \oplus \mathbf{X}_q)$ can recursively find all Φ_p such that $\Phi_p \leq_{\mathcal{Q}} \Phi_q$ and adds no computations applying to any $X \in \mathbf{X}_q$. Also for fixed θ_i, \vec{m}, j the question of whether $T(\Phi_p, \neg\theta_i(\vec{m}), j)$ has an infinite path on which A does not appear can be decided by $(B \oplus C)^{(\alpha+1)}$ (as $T(\Phi_p, \neg\theta_i(\vec{m}), j)$ at least one level of complexity lower than $T(\Phi_p, \psi, j)$) and so $(B \oplus C)^{(\alpha+1)} \oplus \mathbf{X}_q$ suffices to find such a $\Phi_p, \theta_i, \vec{m}$ and j and to construct a path Y in the tree, and so $r = (\Phi_p, \mathbf{X}(Y) \cup \mathbf{X}_q)$ is the condition we want.

Case 3: In the final case $T(\Phi_q, \psi, k)$ is infinite, but every path contains A . We claim that q does not force ψ , because if $q \Vdash_{\mathcal{Q}} \psi$, then by Lemma 4.2, \mathbf{X}_q forms a branch of $T(\Phi_q, \psi, k)$ not containing A . As ψ is a universal sentence and q does not force it, there is some $p \leq_{\mathcal{Q}} q$, some conjunct $\theta_i(\vec{u})$ and some tuple \vec{m} such that for every $p' \leq_{\mathcal{Q}} p$, p' does not force $\theta_i(\vec{m})$. Therefore, $p \Vdash \neg\theta_i(\vec{m})$.

For such an p there is a j such that $T(\Phi_p, \neg\theta_i(\vec{m}), j)$ has a path Y on which A does not appear, and so by Corollary 4.4 $(B \oplus C)^{(\alpha+1)}$ can find such a $\Phi_p, \theta_i, \vec{m}$ and then construct a path Y , so then, the condition we want is $r = (\Phi_p, \mathbf{X}_q \cup \mathbf{X}(Y))$. \square

The reader may be wondering why we are proving theorems which are the analogues of theorems which prevent us from adding A to the infinite part of a \mathcal{P} condition, when all \mathcal{Q} conditions have that built in. It is, in fact, the bounds on the complexity of the forcing relation that we want. These bounds will allow us to diagonalize against computing D .

Definition 4.6 ($(\mathcal{Q}, B \oplus C)$ -Essential to splits). Let Φ_0 be a finite use-monotone Turing functional which correctly computes B on input A , and $\langle e, \alpha \rangle$ a $(B \oplus C)$ -hyperarithmetic reduction. For each τ , a finite sequence of binary strings of the same length, we say τ is $(\mathcal{Q}, B \oplus C)$ -**essential to $\langle e, \alpha \rangle$ -splits below Φ_0** if whenever $q, r \leq_{\mathcal{Q}} (\Phi_0, \emptyset)$ form an $\langle e, \alpha \rangle$ -split (i.e. there is an x such that $q \Vdash_{\mathcal{Q}} \langle e, \alpha \rangle^{B \oplus C \oplus \Phi_G}(x) \downarrow = k_1, r \Vdash_{\mathcal{Q}} \langle e, \alpha \rangle^{B \oplus C \oplus \Phi_G}(x) \downarrow = k_2$ and $k_1 \neq k_2$) there is some new axiom $\langle x, y, \sigma \rangle \in \Phi_q \cup \Phi_r$ but not in Φ_0 such that σ is compatible with some component of τ .

We let $U_{\mathcal{Q}, B \oplus C}(\Phi_0, \langle e, \alpha \rangle, k)$ denote the set of length k vectors τ which are $(\mathcal{Q}, B \oplus C)$ -essential to $\langle e, \alpha \rangle$ -splits below Φ_0 . As with $U_{\mathcal{P}, C}(\Phi_0, \langle e, \alpha \rangle, k)$, this set can be considered as a subtree of a recursively bounded recursive tree. We drop the \mathcal{Q} and $B \oplus C$ decorations wherever possible.

Lemma 4.7. *Let Φ_0 be a Turing functional which partially computes B on input A , $\langle e, \alpha \rangle$ a $B \oplus C$ -hyperarithmetic reduction, and k a natural number.*

- *If \mathbf{X} is a size k set such that $(\Phi_0, \mathbf{X}) \in \mathcal{Q}$, and no pair $q, r \leq_{\mathcal{Q}} (\Phi_0, \mathbf{X})$ form an $\langle e, \alpha \rangle$ -split, then $U(\Phi_0, \langle e, \alpha \rangle, k)$ has an infinite path on which A does not appear.*
- *If $U(\Phi_0, \langle e, \alpha \rangle, k)$ has an infinite path on which A does not appear, then each such path Y is identified with a size k set $\mathbf{X}(Y)$ such that $(\Phi_0, \mathbf{X}(Y)) \in \mathcal{Q}$ has no $\langle e, \alpha \rangle$ -splits below it.*

Proof. The strategy is similar to that of Lemma 3.8, with the same modifications needed to turn a proof of Lemma 3.3 into one of Lemma 4.2. \square

Lemma 4.8. *If Φ_0 is a Turing functional which partially computes B on input A , $\langle e, \alpha \rangle$ a $(B \oplus C)$ -hyperarithmetic reduction, and k is a natural number, then $U(\Phi_0, \langle e, \alpha \rangle, k)$ is $\Pi_{\alpha+3}(B \oplus C)$, uniformly in the defining parameters. Furthermore if there exists a size k set \mathbf{X} , such that $(\Phi_0, \mathbf{X}) \in \mathcal{Q}$ has no $\langle e, \alpha \rangle$ -splits below it, then we can find such an \mathbf{X} not containing A all of whose members are recursive in $(B \oplus C)^{(\alpha+4)}$ (it is also uniform, although this is not needed).*

Proof. The strategy is similar to that of Lemmas 3.9 and 3.10. Fix $\Phi_0, k, \langle e, \alpha \rangle$ and τ . Then $\tau \in U(\Phi_0, \langle e, \alpha \rangle, k)$ iff

$$\begin{aligned}
& (\forall \Phi_q, \Phi_r \leq_{\mathcal{Q}} \Phi_0) (\forall m, m_1, m_2) [(\exists \mathbf{X}_q, A \notin \mathbf{X}_q) ((\Phi_q, \mathbf{X}_q) \Vdash \langle e, \alpha \rangle^{(B \oplus C \oplus \Phi_G)}(m) \downarrow = m_1) \\
& \quad \wedge (\exists \mathbf{X}_r, A \notin \mathbf{X}_r) ((\Phi_r, \mathbf{X}_r) \Vdash \langle e, \alpha \rangle^{(B \oplus C \oplus \Phi_G)}(m) \downarrow = m_2) \\
& \quad \wedge m_1 \neq m_2) \\
& \rightarrow (\exists \langle x, y, \sigma \rangle) \text{ such that } \langle x, y, \sigma \rangle \in (\Phi_q \cup \Phi_r) \setminus \Phi_0 \\
& \quad \text{yet } \sigma \text{ is compatible with some } \tau \in \tau]
\end{aligned}$$

By Lemma 4.7 we can replace the existential quantifiers over reals by

$(\exists j)U(\Phi_q, \langle e, \alpha \rangle^{(B \oplus C \oplus \Phi_G)}(m) \downarrow = m_1, j)$ has a path on which A does not appear.

(making the analogous changes for Φ_r). Now, the formula “ $\langle e, \alpha \rangle^{(B \oplus C \oplus \Phi_G)}(m) \downarrow = m_1$ ” can be rendered as a $\Sigma_{\alpha+1}^r(B \oplus C)$ formula in our forcing language so, increasing its level by 1 to get a $\Pi^r(B \oplus C)$ formula, and applying Lemma 4.3, the existential quantifier over reals is equivalent to a (uniformly given) $\Pi_{\alpha+2}^0(B \oplus C)$ tree having a path on which A does not appear.

We can decide whether the tree has such a path by answering the question: Does there exist a j and a node on a uniformly given $\Pi_{\alpha+2}^0(B \oplus C)$ subtree of a recursively bounded tree, the coordinates of which all disagree with A , above which the tree is infinite? Which is a uniformly $\Sigma_{\alpha+3}^0(B \oplus C)$ question.

This means we can express “ τ is essential” as a formula with a prefix of universal quantifiers, then a uniformly given $\Pi_{\alpha+3}^0(B \oplus C)$ matrix, and so this property is $\Pi_{\alpha+3}^0(B \oplus C)$, as required.

So, to find a “simple” \mathbf{X} , provided there is one, we need to be able to construct the tree $U(\Phi_0, \langle e, \alpha \rangle, k)$ and find paths on it, on which A does not appear. As $U(\Phi_0, \langle e, \alpha \rangle, k)$ is a $\Pi_{\alpha+3}^0(B \oplus C)$ subtree of a recursively bounded recursive tree $(B \oplus C)^{(\alpha+4)}$ can decide whether a given τ is on the tree and if so whether the tree is infinite above that node. It can also decide whether the node has a coordinate which is an initial segment of A , and so can construct a the desired path provided one exists. \square

Corollary 4.9. *Let $q \in \mathcal{Q}$ be a forcing condition and $\langle e, \alpha \rangle$ a $(B \oplus C)$ -hyperarithmetical reduction. Then we can find an extension $r \in \mathcal{Q}$ of q which diagonalizes against $\langle e, \alpha \rangle^{B \oplus C \oplus \Phi_G} = D$ i.e. p either forces $\langle e, \alpha \rangle^{B \oplus C \oplus \Phi_G}$ is not total, or, there is an x such that it forces this computation on x to be convergent, yet not equal to $D(x)$.*

Proof. The idea is the same as in Corollary 3.11. We assume there is a condition q which forces totality and has no $\langle e, \alpha \rangle$ -splits. We then move to the related condition q' which has the same finite part as q , but the real parts are all recursive in $(B \oplus C)^{(\alpha+4)}$ as guaranteed by Lemma 4.8. We then search for extensions of q' forcing $\langle e, \alpha \rangle^{B \oplus C \oplus \Phi_G}(x) \downarrow = y$ and output y if we find it. We only need to search for conditions with new reals recursive in $(B \oplus C)^{(\alpha+1)}$ by Corollary 4.5. This makes the search hyperarithmetical in $B \oplus C$, and so, if it computes D , then $B \oplus C$ computes D , contrary to our hypothesis. \square

Corollary 4.10. *The Σ_2 theory of the arithmetic degrees in \leq, \sqcup and 0 is decidable.*

Proof. By Jockusch and Slaman, it suffices to establish the extended Posner-Robinson theorem. Choose representatives for each of the arithmetic degrees $\mathbf{a}, \mathbf{b}, \mathbf{c}_i, \mathbf{d}_i$ such that if i is a hard case then $A \leq_T C_i$.

One constructs a sequence (beginning with (\emptyset, \emptyset)) of conditions (Φ_i, \mathbf{X}_i) maintaining that Φ_i partially computes B given A and that $A \notin \mathbf{X}_i$. Hence, our sequence will live in both \mathcal{P} and \mathcal{Q} . We dovetail together all the Π_n^r formulas for $n < \omega$, and relevant arithmetic reductions against which we will diagonalize, and successively extend our sequence in the

corresponding notion of forcing (\mathcal{P} for easy cases, \mathcal{Q} for hard) deciding the next sentence, and diagonalizing against the reductions. This ensures that forcing equals truth for all arithmetic sentences of our forcing languages. By Corollary 3.6 and 3.10 we can do this for the easy cases without adding A to the infinite part or incorrect axioms applying to A to the finite part, as in the easy case A is not arithmetic in C .

Between these stages we add the axiom $(n, B(n), \alpha)$ for the least n so that there is no axiom applying to A of this form, and α is the shortest initial segment of A permitted to enter. As we are maintaining $A \notin \mathbf{X}_i$ this is possible, and we never add any incorrect axioms applying to A . This ensures that $A \oplus \Phi_G \geq_a B$ as required.

As we have diagonalized against each of the listed arithmetic reduction and forcing equals truth for each arithmetic formula, we know for each index e and natural number n that $\langle e, n \rangle^{C_i \oplus \Phi_G} \neq D_i$ in the easy case, and $\langle e, n \rangle^{B \oplus C_j \oplus \Phi_G} \neq D_j$ in the hard case (this relies on the observation that our algorithms in Corollaries 3.11 and 4.9 are arithmetic in the parameters when α is finite). For avoiding the ideals below \mathbf{e}_i , we could do more forcing arguments. For brevity though, we appeal to the fact that there are continuum many generics, even with our coding of B . \square

5 Preserving ω_1^{CK}

To move the previous result to the hyperarithmetic setting we need to show that we can also preserve ω_1^{CK} relativised to C in the easy case, and $B \oplus C$ in the hard case. We use a method suggested by Slaman: forcing over nonstandard models of ZFC.

Definition 5.1. Let \mathcal{M} be an ω -model of ZFC. Then $\mathcal{P}_{\mathcal{M}}$ is the collection of all Kumabe-Slaman conditions $p = (\Phi_p, \mathbf{X}_p) \in \mathcal{P}$ such that every real $X \in \mathbf{X}_p$ is in \mathcal{M} . Because \mathcal{M} is an ω -model this definition is meaningful. Similarly $\mathcal{Q}_{\mathcal{M}}$ is the restriction of the forcing $\mathcal{Q} = \mathcal{Q}_{A,B}$ to \mathcal{M} . Extension is defined as for \mathcal{P} and \mathcal{Q} respectively. Note that $\mathcal{P}_{\mathcal{M}} \in \mathcal{M}$ for any such \mathcal{M} , and $\mathcal{Q}_{\mathcal{M}} \in \mathcal{M}$ if A and B are also members of \mathcal{M} .

Notation. Given a Kumabe-Slaman condition $p = (\Phi_p, \mathbf{X}_p) \in \mathcal{P}$ we call $(\Phi_p, \mathbf{X}_p \cap (2^\omega)^{\mathcal{M}})$ the restriction of p to \mathcal{M} and denote it $p \upharpoonright_{\mathcal{M}}$.

5.1 Easy case

Suppose $C \in \mathbf{c}$ corresponds to an easy case and \mathcal{M} is a countable ω -model of ZFC that omits ω_1^C and A but contains C . The existence of such a model is shown in [2].

Lemma 5.2. *If $(p_i)_{i=0}^\infty$ is an \mathcal{M} -generic sequence (i.e. the sequence meets all of the dense subsets of $\mathcal{P}_{\mathcal{M}}$ that appear in \mathcal{M}), then Φ_G preserves ω_1^C , that is to say, $\omega_1^{C \oplus \Phi_G} = \omega_1^C$.*

Proof. Suppose $<$ is a well-order recursive in $C \oplus \Phi_G$. As \mathcal{M} is an ω -model and the ordinals of $\mathcal{M}[\Phi_G]$ are the same as that of \mathcal{M} , then $\mathcal{M}[\Phi_G]$ is an ω -model containing $C \oplus \Phi_G$ and is closed under Turing reduction. Consequently, $\mathcal{M}[\Phi_G]$ contains an isomorphic copy of $<$, and as $<$ is externally well-founded, it is well-founded in $\mathcal{M}[\Phi_G]$.

Hence, $\mathcal{M}[\Phi_G]$ thinks $<$ is isomorphic to an ordinal α , and as $<$ is externally well-founded, so too is α . However, the only externally well-founded ordinals in $\mathcal{M}[\Phi_G]$ are those that are also externally well-founded ordinals in \mathcal{M} , and by hypothesis, those are precisely the ordinals less than ω_1^C . Therefore, $<$ has height less than ω_1^C and so $\omega_1^{C \oplus \Phi_G} = \omega_1^C$ as required. \square

Lemma 5.3. *Suppose $\mathcal{D} \subseteq \mathcal{P}_{\mathcal{M}}$ is dense, and also an element of \mathcal{M} . Further suppose $p = (\Phi_p, \mathbf{X}_p) \in \mathcal{M}$ is a forcing condition. Then we can find an extension $q \in \mathcal{P}_{\mathcal{M}}$ of p meeting \mathcal{D} such that q adds no new computations applying to A , and A is not in the infinite part of q (as $A \notin \mathcal{M}$).*

Proof. Suppose we can't meet \mathcal{D} below p without adding new computations to A . We will show that \mathcal{D} is not dense. Let τ be a finite sequence of finite binary strings all of the same length. We say τ is essential to meeting \mathcal{D} over Φ_p if for every condition $r = (\Phi_r, \mathbf{X}_r) \in \mathcal{D}$ with $r < (\Phi_p, \emptyset)$, there is an axiom $\langle x, y, \sigma \rangle \in \Phi_r \setminus \Phi_p$ with σ compatible with some component of τ .

Let $V(\Phi_p, \mathcal{D}, k)$ denote the set of τ of length k which are essential to meeting \mathcal{D} over Φ_p and note $V(\Phi_p, \mathcal{D}, k) \in \mathcal{M}$ for all k . Let $k = |\mathbf{X}_p|$ and enumerate $\mathbf{X}_p = (X_1, \dots, X_k)$. We claim $\tau_l = (X_1 \upharpoonright l, \dots, X_k \upharpoonright l, A \upharpoonright l) \in V(\Phi_p, \mathcal{D}, k+1)$ for each l . To see this, suppose $r = (\Phi_r, \mathbf{X}_r) \in \mathcal{P}_{\mathcal{M}}$ properly extends (Φ_p, \emptyset) and meets \mathcal{D} . If r has no new axioms applying to any of X_1, \dots, X_k, A , then r extends p , contradicting our hypothesis that we can't meet \mathcal{D} below p without adding axioms about A . Hence, such an r has a new axiom $\langle x, y, \sigma \rangle$ applying to one of X_1, \dots, X_k, A and so σ is compatible with the corresponding component of τ_l , showing τ_l is essential.

Consequently, $V(\Phi_p, \mathcal{D}, k+1)$ is an infinite finitely-branching tree that is an element of \mathcal{M} . As \mathcal{M} is an ω -model it sees that $V(\Phi_p, \mathcal{D}, k+1)$ is infinite and finitely-branching, so by König's Lemma inside \mathcal{M} , $V(\Phi_p, \mathcal{D}, k+1)$ has a branch Y in \mathcal{M} . This branch corresponds to reals $\mathbf{X}(Y)$ which are also in \mathcal{M} and so $q = (\Phi_p, \mathbf{X}_p \cup \mathbf{X}(Y)) \in \mathcal{P}_{\mathcal{M}}$ and extends p .

As \mathcal{D} is dense there is some extension $r = (\Phi_r, \mathbf{X}_r)$ of q in \mathcal{D} . Such an r extends (Φ_p, \emptyset) so by the definition of essentiality, there is some axiom $\langle x, y, \sigma \rangle \in \Phi_r \setminus \Phi_p$ with σ compatible with some component of each element of Y . Therefore, σ is an initial segment of some $X \in \mathbf{X}(Y)$, but the axiom $\langle x, y, \sigma \rangle$ can only be in an extension of $q = (\Phi_p, \mathbf{X}_p \cup \mathbf{X}(Y))$ if it was already in Φ_p , a contradiction. Therefore, q has no extensions in \mathcal{D} and so \mathcal{D} is not dense. \square

5.2 Hard case

If $A \leq_h C$, then we can't omit A from ω -models of ZFC containing C . Consequently, we can't rely on the fact that $A \notin \mathcal{M}$ to prevent us from adding A to the infinite part of a condition. Indeed, if \mathcal{M} contains A , then $\mathcal{D} = \{p \in \mathcal{P} : A \in \mathbf{X}_p\}$ is a dense subset of $\mathcal{P}_{\mathcal{M}}$ which is also a member of \mathcal{M} . Consequently, we have to use a different forcing.

Of course given what has come before, we want to use $\mathcal{Q}_{\mathcal{M}}$. Any ω -model which contains C , must contain A in the hard case, and consequently, $\mathcal{Q}_{\mathcal{M}}$ is an element of such a model

iff B is. Therefore, in the hard case, we pick an ω -model \mathcal{M} which contains C and B , yet omits $\omega_1^{B \oplus C}$, again this follows from [2].

Once one has such a model, the argument for any set forcing preserving the least omitted ordinal (in this case $\omega_1^{B \oplus C}$) is the same as above. By definition of the forcing we need not worry about interfering with our coding procedure.

5.3 The compatibility lemma

For a condition $p = (\Phi_p, \mathbf{X}_p)$ in either of the restricted forcings $\mathcal{P}_{\mathcal{M}}$ and $\mathcal{Q}_{\mathcal{M}}$, it is still a fact that no extension of p (in the relevant forcing) can add new axioms about members of \mathbf{X}_p . However, as the class of reals from which we can pick members of \mathbf{X}_p from is highly restricted, we want to know we can meet dense sets without adding axioms applying to reals not appearing in the models (to ensure compatibility with the unrestricted forcings).

Lemma 5.4. *Suppose C is an easy case, then for every countable ω -model of ZFC \mathcal{M} omitting ω_1^C and A yet containing C , every finite set of reals \mathbf{X} , every condition $p = (\Phi_p, \mathbf{X}_p) \in \mathcal{P}_{\mathcal{M}}$, and every dense $\mathcal{D} \subseteq \mathcal{P}_{\mathcal{M}}$ which is also an element of \mathcal{M} , there is an extension $r \in \mathcal{P}_{\mathcal{M}}$ of p meeting \mathcal{D} which adds no new computations to any member of \mathbf{X} .*

On the other hand if C is a hard case, then for every countable ω -model of ZFC \mathcal{M} omitting $\omega_1^{B \oplus C}$ yet containing B and C (and, consequently, A), every finite set of reals \mathbf{X} not containing A , every condition $q \in \mathcal{Q}_{\mathcal{M}}$, and every dense $\mathcal{D} \subseteq \mathcal{Q}_{\mathcal{M}}$ which is also an element of \mathcal{M} , there is an extension $r \in \mathcal{Q}_{\mathcal{M}}$ of q meeting \mathcal{D} which adds no new computations to any member of \mathbf{X} .

Proof. For the first claim fix such an \mathcal{M} , an \mathbf{X} , a $p \in \mathcal{P}_{\mathcal{M}}$, and a $\mathcal{D} \in \mathcal{M}$. If any $X \in \mathbf{X}$ is also an element of \mathcal{M} , then the condition $p' = (\Phi_p, \mathbf{X}_p \cup (\mathbf{X} \cap \mathcal{M})) \in \mathcal{P}_{\mathcal{M}}$ extends p and can add no new computations. By the density of \mathcal{D} we can meet \mathcal{D} below p' adding no computations to any $X \in \mathbf{X} \cap \mathcal{M}$, so we assume that \mathbf{X} is disjoint from \mathcal{M} .

Now suppose that every extension of p in \mathcal{D} adds a new axiom applying to some $X \in \mathbf{X}$. As in Lemma 5.3 consider the tree $V(\Phi_p, \mathcal{D}, |\mathbf{X}|)$ of essential τ of length $|\mathbf{X}|$. By assumption $\tau_l = (X_1 \upharpoonright l, \dots, X_n \upharpoonright l)$ is essential to meeting \mathcal{D} below p (where $\mathbf{X} = \{X_1, \dots, X_n\}$). Consequently, $V(\Phi_p, \mathcal{D}, |\mathbf{X}|)$ is a finitely branching tree in \mathcal{M} which is infinite. Hence, it has a path Y in \mathcal{M} corresponding to reals $\mathbf{X}(Y)$ also in \mathcal{M} .

We claim $(\Phi_p, \mathbf{X}(Y))$ has no extensions in \mathcal{D} . If it did, such an r has a new axiom compatible with some component of each member of Y . Consequently, r has a new axiom applying to some $X \in \mathbf{X}(Y)$ and so r can't extend $(\Phi_p, \mathbf{X}(Y))$ a contradiction.

The proof for the second claim in the lemma is similar: as we are assuming \mathbf{X} does not contain A , we can still use the same trick as before to assume \mathbf{X} is disjoint from \mathcal{M} . The proof from there is the same. \square

6 Bringing it all together

Fix representatives $A \in \mathbf{a}$ and $B \in \mathbf{b}$. Then for each $\mathbf{c}_i, \mathbf{d}_i$ pair, pick representatives C_i and D_i , with the requirement that if i is a hard case, then $A \leq_T C_i$. Then, for each easy case

i , fix a countable ω -model \mathcal{M}_i of ZFC containing C_i yet omitting $\omega_1^{C_i}$ and A , and for each hard case j , fix a countable ω -model \mathcal{M}_j of ZFC containing B and C_j yet omitting $\omega_1^{B \oplus C_j}$.

Now we need to show that we can construct a generic for all the forcings simultaneously. We can't construct a single generic sequence, because we may add a real X to the infinite part at some point, where that X is not a member of some \mathcal{M}_i . To work around this we use the Compatibility Lemma.

We will have one master sequence $\{(\Phi_n, \mathbf{X}_n)\}_{n=0}^\infty$ which will live in \mathcal{P} the unrestricted forcing. We will maintain that $A \notin \mathbf{X}_n$, Φ_n partially computes B on input A and $p_{n+1} \leq_{\mathcal{P}} p_n$. These three conditions will guarantee that our sequence is also a descending sequence of \mathcal{Q} conditions. The master sequence will be used to decide all the sentences of the languages $\mathcal{L}_{\omega_1, \omega}^r(C_i)$ and $\mathcal{L}_{\omega_1, \omega}^r(B \oplus C_j)$ for easy cases i and hard cases j . We will also diagonalize against computing D_i . This part is the same as the argument for the arithmetic degrees, and Corollaries 3.6, 4.5, 3.11 and 4.9 say we can build such a sequence.

To meet the dense sets in one of our ω -models \mathcal{M}_i , we consider the sequence of restricted conditions $(\Phi_n, \mathbf{X}_n) \upharpoonright \mathcal{M}_i$. Certainly at some stage of the construction, we can extend the condition $(\Phi_n, \mathbf{X}_n) \upharpoonright \mathcal{M}_i$ in the corresponding forcing living in \mathcal{M}_i to meet a dense set, however, we can do better.

By lemma 5.4, we can extend $(\Phi_n, \mathbf{X}_n) \upharpoonright \mathcal{M}_i$ in \mathcal{M}_i to meet \mathcal{D} , without adding any axioms to A or any member of \mathbf{X}_n , nor adding A to the infinite part. Call such an extension $q = (\Phi_q, \mathbf{X}_q)$. Then we would define the next member of the master sequence to be $p_{n+1} = (\Phi_q, \mathbf{X}_q \cup \mathbf{X}_{p_n})$. By hypothesis $p_{n+1} \leq_{\mathcal{P}} p_n$, p_{n+1} partially computes B given A , and doesn't have A in the infinite part. Thus, we can continue coding, and have maintained our invariants.

Thus, our master sequence (Φ_n, \mathbf{X}_n) decides each sentence of each $\mathcal{L}_{\omega_1, \omega}^r(C_i)$ and $\mathcal{L}_{\omega_1, \omega}^r(B \oplus C_j)$, and diagonalizes against every C_i or $B \oplus C_j$ -hyperarithmetic reduction, depending on the case. The restricted sequence $(\Phi_n, \mathbf{X}_n) \upharpoonright \mathcal{M}_i$ is \mathcal{M}_i generic, and corresponds to the same generic object as for the master sequence. Hence, Φ_G preserves each $\omega_1^{C_i}$ and $\omega_1^{B \oplus C_j}$, which completes the proof of the extended Posner-Robinson theorem.

References

- [1] Ash, C; Knight, J
Computable Structures and the Hyperarithmetical Hierarchy
Studies in Logic and the Foundations of Mathematics Vol. 144, Elsevier Science, 2000
- [2] Harrington L; Shore, R; Slaman, T
 Σ_1^1 in every real in a Σ_1^1 class of reals is Σ_1^1
forthcoming
- [3] Jockusch, C; Slaman, T
On the Σ_2 -theory of the upper semilattice of Turing degrees
The Journal of Symbolic Logic Vol. 58, Number 1, Association for Symbolic Logic, 1993
- [4] Kjos-Hanssen, B; Shore, R
Lattice initial segments of the hyperdegrees
The Journal of Symbolic Logic Vol. 75, Association for Symbolic Logic, 2010
- [5] Lerman, M
Degrees of unsolvability
Perspectives in Mathematical Logic
Omega Series, Springer-Verlag, Berlin, Heidelberg, New York, Tokyo, 1983
- [6] Odifreddi, P
Forcing and Reducibilities
The Journal of Symbolic Logic
Vol. 48, Number 2, Association for Symbolic Logic, 1983
- [7] Sacks, G
Higher Recursion Theory
Perspectives in Mathematical Logic
Springer-Verlag, Berlin, 1990
- [8] Shore, R
The Turing Degrees: An Introduction
Forcing, Iterated Ultrapowers, and Turing Degrees, Lecture Notes Series
Institute for Mathematical Sciences, National University of Singapore, 29, World Scientific Publishing, 2015, 39-121.
- [9] Shore, R; Slaman, T
Defining the Turing Jump
Mathematical Research Letters
6(6):711-722, 1999.
- [10] Simpson, M
Arithmetic Degrees: Initial Segments, ω -REA Operators and the ω -jump
PhD Thesis, Cornell University